DELFT UNIVERSITY OF TECHNOLOGY

MASTERS THESIS

Reproducing a deep learning algorithm - when unstoppable expectations meet immovable reality

---

Author:
SRINATH Jayaraman Nagamani

Supervisor:
Dr. CHRISTOPH Lofi

A thesis submitted in fulfillment of the requirements
for the degree of MSc. Computer Science

in the

Web Information Systems Group
Software Technology

November 23, 2022

"I have only made this ~~letter~~ *thesis longer because I have not had the time to make it shorter."*

~~Blaise Pascal, legendary French mathematician~~
Srinath Jayaraman, (relatively) unknown grad student

DELFT UNIVERSITY OF TECHNOLOGY

Abstract

Electrical Engineering, Mathematics and Computer Science
Software Technology

MSc. Computer Science

Reproducing a deep learning algorithm - when unstoppable expectations meet immovable reality

by SRINATH Jayaraman Nagamani

Investigating changes in forest cover has been an area of intense research for decades. From manual surveys to remote sensing we have come a long way in mapping the world around us. Machine learning and its' younger sibling, deep learning, have emerged as highly useful tools on this journey. There are a wide variety of algorithms that take a stab at analysing vegetation coverage, but not all of them are easily accessible to the community at large. This was the underlying incentive for the paper you are about to read - reproducing and replicating a novel deep-learning algorithm that generates maps of forest cover in a relatively under-explored part of the world. This study lays out the methodology devised by the original authors as well as our attempts to replicate their work on a different set of data. We present the results we were able to obtain, the hurdles we encountered along the way, and a set of guidelines we feel would be helpful for future researchers.

Thesis Committee:
Chair: Associate Prof.dr. Maria Soledad Pera, TU Delft
University supervisor: Assistant Prof.dr. Christoph Lofi, TU Delft
Committee Member: Associate Prof.dr. Cynthia Liem, TU Delft
Committee Member: Dr. Lixia Chu, post-doctoral researcher, TU Delft

Acknowledgements

My masters thesis is the most challenging undertaking I have ever embarked on. Completing it marks the end of a very turbulent period in my life. I have battled severe mental health issues and financial duress for the duration of this thesis. And yet 5 years from now, I might retrospectively be thankful for the many valuable lessons learned about consistency, hard work, and most importantly - acknowledging the fact that we all need help every now and then.

This endeavour would not have been possible without my supervisor Dr. Christoph Lofi. His patience and confidence in my abilities gave me the push I needed to do this. I have enjoyed our interactions immensely and his guidance will be one of the fondest memories from my time at TU Delft. I would also like to thank Dr. Lixia Chu, a postdoctoral researcher in the Web Information Systems research group for her invaluable inputs that got me out of quite a few sticky situations! I would also like to extend my warmest thanks to the rest of my committee, Dr. Cynthia Liem and Dr. Maria Soledad Pera for making time to evaluate my work.

I would also like to thank my family. Without their moral and financial support I would've never had the self-belief to complete a masters degree.

No list of acknowledgements would be complete without mentioning my older sister, Dr. Subhadra Jayaraman. Her boundless faith in my potential was inspiring beyond belief, and it helped me recognise my own capabilities.

Moreover, I must express profound gratitude toward my friends Bhoomika, Akash, Simran, Ashika, Michael, and Neha. Writing a masters thesis is a gruelling task by itself, but doing it against the backdrop of a global pandemic was tenfold more stressful. My friends made the experience far more enjoyable than it had any right to be and I will forever be grateful to them for it.

Last but not the least, I'd like to give a big shout-out to Chaleo Yoovidhya and Dietrich Mateschitz, the businessmen responsible for creating Red Bull. Without it, typing a masters thesis would've been a burden of Sisyphean proportions.

Srinath Jayaraman
November 2022

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The increasing accessibility of satellite data (aka remote sensing data) has spurred an ever-increasing interest in research and analysis of remote sensing data. One of these areas of research is **change detection**, which is defined as "*the process of identifying differences in the state of an object or phenomenon by observing it at different times*"[85]. In 2008 the United States Geological Survey (USGS) opened the Landsat archive to the public [53]. Predictably, this triggered a whole new wave of research that employed multi-spectral and multi-temporal methods of change detection using remote sensing data. Change detection is employed in a variety of use cases, some of which are given below:

- Land use/land cover (LULC) change detection, as described in [17, 61, 69]

- Urban change detection, as described in [62, 15, 4]

- Natural disaster change detection, as described in [38, 92, 99]

- Forest cover change detection, as described in [101, 44, 48, 103]

Historically, forests were monitored by land-based and/or aerial surveys. These surveys are very time consuming and require a lot of financial resources and significant manpower. Owing to this, developing and under-developed countries often lag behind advanced nations in accurately mapping the extent of forested land within their borders. This in turn causes more problems, as it becomes difficult to establish the true level of deforestation (or afforestation).

Several methodologies have been proposed to monitor forests using satellite imagery. Some approaches are global [91, 52, 82], while some are more localised/regional [18, 51, 32]. Other attempts have focused on a specific kind of forest cover [25, 87] or structure [56]. A recurring theme, however, is bi-temporal change detection or statistical machine learning techniques to detect changes. According to [82] "In all global mapping efforts and the majority of the others, data from optical sensors have been used and maps have been generated for a **single year or period**, largely because of requirement of multiple acquisitions to obtain cloud-free images."

For the purposes of this thesis, we will focus on replicating one existing DL-based forest cover change detection framework called. Replicating and comparing *multiple* DL frameworks is out of the scope of this document. In subsequent sections we will discuss the criteria behind choosing AI-ForestWatch for replication, what would constitute a successful replication, and define our research question and contributions.

1.1   Selection criteria

Given the complexity and enormous heterogeneity of papers exploring the idea of DL-based forest change detection, choosing just one to replicate and reproduce was a tough ask. To simplify this task (at least a little bit), we formulated a set of "minimum" criteria predicated on the conclusions arrived upon in the papers referenced above:

1. Analysis carried out in a relatively less explored part of the world;

2. Code was publicly available (preferably on GitHub);

3. Data was publicly available;

4. Data was present in a **non-proprietary** format; and

5. No paid software was used for any task, no matter how insignificant

Papers had to meet **all the requirements** listed above to be considered. The reasoning being that it should be as easy as possible for researchers (or grad students) to replicate a software based study. Since running code authored by someone else is *already* a tricky proposal, we attempted to eliminate or at the very least, minimise additional barriers further hindering reproducibility and replicability. But selecting one paper to replicate was trickier than we anticipated. There are a lot of papers published in the area of forest change detection, but a lot of them do not have either publicly available data or code, and sometimes both are not available. Table 1.1 shows a sample of the papers that were considered and the status of their data and code availability.

Title	Data available?	Code available?
AI-ForestWatch: semantic segmentation based end-to-end framework for forest estimation and change detection using multi-spectral remote sensing imagery	Yes	Yes
Automated prediction system for vegetation cover based on MODIS-NDVI satellite data and neural networks	Yes	No
Proposal of Prediction Technique for Future Vegetation Information by Climate Change using Satellite Image	No	No
Satellite Image Prediction Relying on GAN and LSTM Neural Networks	Yes	No
MCSIP Net: Multichannel Satellite Image Prediction via Deep Neural Network	No	No
Combined Use of Multi-Temporal Optical and Radar Satellite Images for Grassland Monitoring	No	No
Object-based multi-temporal and multi-source land cover mapping leveraging hierarchical class relationships	Yes	No
Combining Sentinel-1 and Sentinel-2 Satellite Image Time Series for land cover mapping via a multi-source DL architecture	Yes	No
Vegetation cover estimation using convolutional neural networks	No	No

TABLE 1.1: Some papers considered for replication along with their data and code availability

Based on these criteria, we chose the DL-based forest cover change detection framework **AI-ForestWatch**[103] for reproduction and replication. In it, the authors introduce an original end-to-end framework that "*uses deep convolution neural*

network-based semantic segmentation to process multi-spectral space-borne images to quantitatively monitor the forest cover change patterns by automatically extracting features from the dataset"[103].

In simpler terms, a slightly modified CNN, called a UNet, is used to scrutinise multi-spectral *(covering multiple sections of the electromagnetic spectrum)* and multi-temporal *(covering multiple years from 2014 to 2020)* images from the Landsat-8 satellite, with the goal of generating forest cover change maps for several years at a time.

Before proceeding further, in Section 1.1.1 below, we provide a brief explanation of the difference between **reproduction** and **replication** to ensure clarity for the reader.

1.1.1 Reproduction vs. replication

Although these 2 terms are used interchangeably in the common vernacular, they carry specific meaning in academia. Peng[76] provides a simple and lucid definition for both:

1. **Reproduction** - Analysis of an existing study to see if we get the same results using the same methodology on the same set of data as the original publication.

2. **Replication** - Analysis of an existing study to see if we get the same results using the same methodology *on a different set of data*.

1.2 Research questions

We outline certain conditions that would constitute successful reproduction and replication of AI-ForestWatch.

- Results of the original paper have to be re-created on the original input files:
  - Yearly forest **cover** maps for each district under study (from 2014 to 2020)
  - Temporal forest **change** maps for each district under study

- Results have to be re-created on new input files (for The Netherlands) that the algorithm has not yet encountered.

Based on the criteria listed above, we can then define our objective/research questions as follows:

1. Are the results presented in our chosen DL research paper fully reproduced and replicated?

2. If not, then what factors could have contributed to an unsuccessful reproduction or replication?

3. What proposals (if any) can we present for improving reproducibility?

1.3   Contributions

Our contributions are as follows:

1. Reproduce a novel deep learning algorithm that performs forest cover change detection on Landsat-8 satellite images:

   - The original paper analysed 15 districts in Pakistan
   - Source code and data-sets available on GitHub/Google Drive

2. Replicate the algorithm in a different context:

   - We used Landsat-8 images of The Netherlands for this purpose

3. Use the problems we encountered during the replication/reproduction process as a foundation for devising with a simple, programming language-agnostic framework to help ensure better reproducibility for future research in this field

1.4   Thesis outline

The rest of the thesis is organised as follows: in Chapter 2 we present the related work in land cover classification and forest change detection; in Chapter 3 we explain some of the fundamentals of remote sensing; in Chapter 4 we detail the **original** approach/methodology; in Chapter 5 we explain the steps taken to **reproduce** the original paper and the issues encountered therein; in Chapter 6 we detail the process of replicating the AI-ForestWatch framework on our own input data; and finally in Chapter 7 we present the conclusions of our replication study.

Chapter 2

Related work

Hussain et al.[40] classifies the methods of forest change detection as falling under 2 categories:

1. Pixel-based change detection; and

2. Object-based change detection

They're explained in detail in Sections 2.1 and 2.2 below. Section 2.3 gives a little more background on DL-based change detection methods.

2.1 Pixel based change detection

In pixel-based methods of analysing forest cover change, a pixel-wise comparison is carried out on multi-spectral images in order to discern changes between different images [42, 58, 57]. A list of pixel based change detection methods and their corresponding description is given below. All definitions are taken from [40], and have been cited appropriately.

1. "Image differencing - Two precisely co-registered multi-temporal images are used to produce a residual image to represent changes. The difference can be measured directly from radiometric values of the pixel or on the extracted /derived/ transformed images such as texture or vegetation indices. Mathematically, the difference image is:

$$I_d(x, y) = I_1(x, y) - I_2(x, y)$$

Where I_1 and I_2 are images from time t_1 and t_2 and *(x, y)* are coordinates and I_d is the difference image. Pixels with no change in radiance are distributed around the mean [65], while pixels with change are distributed in the tails of the distribution curve [85]. Since change can occur in both directions, it is therefore up to the analyst to decide which image to subtract from which [24]." [40]

2. "Image rationing - A ratio between two co-registered images is computed. Mathematically:

$$I_r(x, y) = \frac{I_1(x, y)}{I_2(x, y)}$$

Unlike in image differencing,the order of the images in the division is not important as the change results are expressed in ratios,and areas that are not changed should theoretically have a value of 1." [40]

3. "Image regression - The image I_2 from time t_2 is assumed to be a linear function of image I_1 from time (t_1). The image I_2 is taken as the "reference" image and I_1 as a "subject" image. The subject image is then adjusted to match the radiometric conditions of the reference image. A regression analysis, such as least-squares regression, can help identify gains and offsets by radio-metrically normalising the subject image to match the reference image [66]. Change I_d is detected by subtracting regressed image from the first-date image." [40]

4. "Vegetation index differencing - Vegetation indices are mathematical transformations designed to evaluate the impact of vegetation on observations in multispectral mode. These indices enhance the spectral differences on the basis of strong vegetation absorbance in the red and strong reflectance in the near-infrared band. For CD, generally, the vegetation indices are produced separately for two images and then standard pixel based CD (e.g. differencing or ratioing) are applied. Different vegetation indices have been developed such as:

   - Ratio based, including Ratio Vegetation Index (RVI) and the Normalized Difference Vegetation Index (NDVI);

   - Orthogonal indices, including Perpendicular Vegetation Index (PVI) and Difference Vegetation Index (DVI); and

   - Soil Adjusted Vegetation Index (SAVI) and modified soil adjusted vegetation index(MSAVI) [10].

$$RVI = \frac{n}{r}$$

$$NDVI = \frac{n - r}{n_r}$$

$$TVI = \sqrt{\frac{n - r}{n + r} + 0.5}$$

$$SAVI = \frac{n - r}{n - r + L}(1 + L)$$

$$MSVI = \frac{2n + 1 - \sqrt{(2n + 1)^2 - 8(n - r)}}{2}$$

   Where n is near infrared band and r is the red band. The L in SAVI confirms the same bound between NDVI and SAVI. [94] modifying RVI and NDVI by calculating angle vegetation index [93] and developed a bi-temporal vegetation Time-Dependent Vegetation Indices (TDVI)." [40]

5. "Change vector analysis (CVA) - It allows simultaneous analysis of multiple image bands for CD. The idea behind CVA is that a particular pixel with different values over time resides at substantially different location in the feature space [45]. The pixel values are treated as vectors of spectral bands and change vector (CV) is calculated by subtracting vectors for all pixels at different dates [68]. The direction of the CV depicts the type of change whereas the magnitude of the change corresponds to the length of the CV. CVA can also be performed on the transformed data (e.g. Kauth-ThomasTransformation, KTT)." [40]

6. "Principal component analysis (PCA) - PCA, mathematically based on "Principal Axis Transformation", is a transformation of the multivariate data to a new set of components, reducing data redundancy [59]. PCA uses either the covariance matrix or the correlation matrix to transfer data to an uncorrelated set. The eigenvectors of the resulting matrices are sorted in decreasing order where first principal component (PC) expresses most of the data variation. The succeeding component defines the next largest amount of variation and is independent (orthogonal) of the preceding principal component." [40]

   "[12] argued to examine the eigen-structure of the data and visual inspection of the combined images to analyze change types. Sometimes, to determine change type, a grouping of values in a PCA ordination plot is done; however, [104] argued that it can be inaccurate or misleading without knowledge of the actual change that has occurred." [40]

7. "Kauth-Thomas Transformation (KT)/Tasseled Cap transformation - The KT is orthogonalization (linear transformation) of a multi-band, and multi-date dataset and differs from PCA in terms that it is fixed. These output features represent the greenness brightness and wetness. Presented by [50] it analyses the structure of the spectral data, which is a function of a particular characteristic of scene classes. Unlike the PCA, MKT is not scene-dependent and uses of stable and calibrated transformation coefficients which ensures that its application is suitable between regions and across time [13]. The change is measured based on the brightness, greenness and wetness values [64]." [40]

8. "Post-classification comparison - It is arguably the most obvious quantitative CD method because it provides from-to change information [6, 41, 46]. Originally used in the late 70s, it compares two classified images to generate a change matrix, it is often used as a benchmark for the qualitative evaluation of emerging CD techniques [66]. In this approach, bi-temporal images are first rectified and classified. The classified images are then compared to measure changes. The classes for both the images have to be identical to enable one-to-one comparison. The errors from individual image classification are propagated in the final change map, reducing the accuracy of the final CD [14, 59, 9]. In order to improve CD results, the classification of individual images has to be as accurate as possible." [40]

9. "The composite or direct multi-date classification - The composite or direct multi-date classification technique is among the earliest semi-automated approaches to generating land-use and land-cover change maps where a single analysis for multi-date data-sets is performed [66]. Multi-temporal and rectified images are first stacked together. PCA technique is often applied to reduce the number of spectral components to a fewer principal components [70, 85]. The minor components in PCA tend to enhance the spectral contrast and represent changes [11]. The temporal and spectral features have equal status in the combined dataset, making it difficult to separate the spectral changes within one multi-spectral image from temporal changes between images in the classification [80]." [40]

10. "Machine Learning - Artificial neural networks (ANN) algorithms for image based CD belong to the classification-based CD category. ANN algorithms are

non-parametric and make no assumptions about data distribution and independency. They adaptively estimate continuous functions from data without specifying mathematically how outputs depend on inputs [41]. ANN algorithm learns from the training dataset and build relationships (networks) between input (image) and output nodes (changes). The trained network then is applied to the main dataset to create a change map [14, 28]. The ANN approach can provide better CD results when land- cover classes are not normally distributed [64].

"The Support Vector Machine (SVM) is a supervised non-parametric statistical learning technique and makes no assumption about the underlying data distribution. The SVM is based on statistical learning theory which implements structural risk minimization for classification [95]. When applied to stacked multi-temporal images, the change and no-change is treated as a binary classification problem [39]. The algorithm learns from training data and automatically finds a threshold values [7] from the spectral features for classifying change from no-change." [40]

"The decision tree (DT) classification algorithms are also non-parametric with no assumption about data distribution and independency. These DT algorithms build a flow-chart-like tree (hierarchical) structure in which each node represents a test on a number of attribute values, each branch represents an outcome of the test, and tree leaves represent classes or class distribution [31, 54]. The classification rules at the node of the DT are based on the analysis of attribute values. Once a DT is built it can be used for classifying the unknown cases. Change vs. no-change can be treated as a binary-classification problem or a post-classification comparison can be performed to measure changes." [40]

11. "GIS based - Most of the current image processing systems are either integrated or compatible with geographic information systems (GIS). GIS provides a base for data integrating, visualizing, analyzing and map producing. The flow of the data can be bidirectional, as GIS data can be used to overlay onto an image; alternatively, the results from image analysis and can be used to update the GIS data. For example, the parcel layers stored in a GIS database are used to assist classification and CD from an image (see e.g. [86]). Similarly, image data is used to update the GIS database." [40]

"The applicability of GIS with RS integration is enhanced by the more frequent use of object-based image analysis techniques. The spatial and aspatial information about objects stored in the GIS database can play an important role when linked to the objects extracted from RS image for CD along with other image analysis [6]. For example, [98] presented an object-based technique for CD where the training data is extracted from the GIS database to classify the image. The classified objects from the images were then compared against the objects stored in GIS to measure changes." [40]

12. "Texture analysis based change detection - Texture features from images are measured and compared for CD. Texture provides information about the structural arrangement of objects and their relationship with respect to their local neighborhoods [8]. Change is measured by comparing the textural values from images. Among several texture measuring algorithms, a common is a greylevel co-occurrence matrix (GLCM) which is a second order statistics [34, 79]. GLCM examine the spectral as well as spatial distribution of grey

values. Rather than per-pixel comparison, the image is normally divided into smaller windows; texture is calculated and comparison is done at window level. [36] emphasized using texture information only in conjunction with spectral data." [40]

13. "Multi-temporal spectral mixture analysis - Spectral mixture analysis (SMA) has been used to address the increased dimensionality (more than one target class in one pixel) because of high spectral resolution. The assumption in SMA is that multi-spectral image pixels can be defined in terms of their sub-pixel proportions of pure spectral components which may then be related to surface constituents in a scene. In a simple case, linear mixture model, end-member (scene element with a spectral response that is indicative of a pure cover type) spectra weighted by the percent ground cover of each end-member are linearly combined [97]. A linear spectral mixture model is given as:

$$r_i = \sum_{j=1}^{n} a_{ij} f_j + e_i$$

r_i = measured reflectance of a given pixel in spectral band i, n is the number fo mixture components, f_j is the areal proportion, or fraction, of endmember j in r_i, a_{ij} is the reflectance of endmember j in spectral band i, and e_i is the residual, the difference between the observed (ri) and modeled pixel values [97]." [40]

14. "Fuzzy change detection - Fuzziness deals with the ambiguity of class labeling and implies that the boundaries between different classes and phenomena are fuzzy and that there is heterogeneity within a class perhaps due to the physical differences [63]. This becomes important when there is difficulty in selecting a threshold valued to distinguish change from no-change. The results of fuzzy reasoning are not discrete and crisp, but are, rather, expressed in terms of 'probabilities' [71]. It can contain elements with only a partial degree of membership. Fuzzy membership differs from probabilistic interpretation as fuzzy set is defined by a membership function (degree of membership) and the class with highest probability is interpreted as actual class. Post-classification comparison can then be applied to measure the change [19, 21, 22, 23]." [40]

15. "Multi-sensor data fusion for change detection - Acquiring RS data at varied spatial, spectral and temporal resolutions formulate an image pyramid that allows getting data at different resolutions. The data from different sensors reflect specific aspects of terrain and using data from different sensors might help identify certain properties. Although working with different sensors is not ideal, it is sometimes useful especially when time series analysis is performed and one of the sensors may not be available [81]. Multispectral RS data is also useful when dealing with heterogeneous land uses and three dimensional structures especially in urban areas [29, 77]." [40]

2.1.1 Limitations of pixel-based methods

Important contextual information, such as geospatial data is often not modeled while performing pixel-based change detection. These methods are also not suitable for analysing VHR (very high resolution) images [41]. This is mainly due to the fact that VHR images have high variability, which results in an excessive

number of changes being detected - this is known as the "salt and pepper" effect [74].

2.2   Object based change detection

In contrast to pixel based detection methods, object based change detection methods rely on assimilating contextual (for e.g., spatial data) and spectral information [5]. Usually, this is done via unsupervised machine learning techniques for object-based approaches. These do perform better since they are able to prevail over spectral variations due to noise. However, they have the downside of sometimes generating imprecise segmentation results [40]. Object-based methods utilise an image object (i.e., image segmentation), which often contain a lot more contextual information than a single pixel (or group of pixels) does - including texture, shape, spatial information, etc.
Similar to pixel-based methods, a list of object-based change detection methods and their corresponding descriptions is given below. All definitions are taken from [40], and have been cited appropriately.

1. "Direct Object change detection (DOCD) - A direct comparison between the image-objects from different dates is performed for CD, which is similar to the pixel-based approaches. Change is detected either by (a) comparing the geometrical properties (width, area and compactness) [55, 102], or (b) comparing spectral information (mean band values) [30], and/or extracted features (e.g. texture) [55, 90] of the image objects." [40]

   "Broadly speaking, two strategies are developed. In the first, objects from image at time t_1 are extracted, and are assigned to or searched from image at time t_2 without segmentation [72]. In the second approach, segments from multi-temporal images are extracted and compared for CD [74]. The disadvantage of the first approach is that change is linked to only the objects extracted from first image and will not provide new objects that might be created in the second image because of change. The second approach, however, allows using all the objects from both images for change analysis." [40]

2. "Classified Objects change detection (COCD) - Perhaps the most commonly used OBCD methodology that allows the creation of a change matrix indicating the "from– to" changes. OBIA is performed on multi-temporal images to extract objects and independently classify them. The classified objects are compared for a detailed change analysis. Objects are compared based on both the geometry and the class membership [16, 35, 47]. A theoretical framework of OBCD based on post-classification comparison was provided by Blaschke (2005) for the comparison of multi-temporal map objects to detect and identify changes."[40]

   "The performance of COCD is strongly related to the performance and accuracy of the classification algorithm, similar to pixel-based approached. The classification accuracy in OBIA is also related to the selection of image segmentation technique which can results objects of different sizes based on different segmentation parameters." [40]

2.3   DL based forest change detection methods

As outlined in [103], there are 2 categories of forest cover classification -
patch-based and pixel-based.
In patch-based classification, the entire multi-spectral image is dissected into equal
size *patches*, for e.g., 64x64 pixels each. Then, a deep learning neural network,
usually a CNN, allocates/assigns just one class to every single patch based on
whichever class happens to have the most dominant distribution in a given patch.
The 2 most well known and oft-cited examples of this method are ResNet [37] and
GoogleNet [89]. Although their performance has been well-documented, they end
up producing low-resolution maps since an entire patch is labelled with one class.

In pixel-based classification methods, a very commonly used technique is utilising
an encoder-decoder topology like the one found in [2] (aka SegNet) and [20]. In
AI-ForestWatch, the authors have used images from 2015 (combined with
vegetation indices) for estimating the forest cover and generated change maps for
all the other years (2014 and 2016 to 2020). Additionally, UNet-based semantic
segmentation is done on patches of the input data (from Landsat-8) in order to
achieve pixel-wise classification. "The advantage of using this approach is that the
high resolution of the image can be fully exploited to generate detailed forest cover
maps at the same spatial resolution." [103]

Chapter 3

GIS and remote sensing fundamentals

Before delving into the technicalities of AI-ForestWatch, we must learn a few crucial terms used in the world of GIS and remote sensing. It's important for us to understand these fundamentals because they're important to understanding the working of AI-ForestWatch itself. The concepts of vector data, raster data, shape-files, and ground truth data are important to understanding what the original paper and our attempts to replicate it are trying to accomplish.

Based on the manner of collecting data, remote sensing can be classified as either **active** or **passive**. Examples of **active** remote sensing are RADAR and Li-DAR, which measure factors such as distance, angle, and radial velocity by emitting radio waves (RADAR) or lasers (Li-DAR) at objects located considerable distances away. In contrast, **passive** remote sensing is the detection and measurement of "radiation emitted by the Sun and reflected or emitted by the Earth."[1]. Landsat-8 falls under passive remote sensing.

3.1   Vector data

Vector and raster data describe the features around us (rivers, houses, trees, etc.) within a GIS (Geographic Information System) environment. In the case of vector data, this is done via **geometric** features - points, polylines, and polygons:

- "A point feature is described by its X, Y and optionally Z coordinate. The point attributes describe the point e.g. if it is a tree or a lamp post."[2]. Shown in Figure 3.1

- "A polyline is a sequence of joined vertices. Each vertex has an X, Y (and optionally Z) coordinate. Attributes describe the polyline."[3]. Shown in Figure 3.2

- "A polygon, like a polyline, is a sequence of vertices. However in a polygon, the first and last vertices are always at the same position."[4]. Shown in Figure 3.3

The reason we need to understand the importance of vector data is because the geographical information about our physical world - coordinates, boundaries, etc. -

---

[1]PennState, Dept. of Geography
[2]Point Features - QGIS documentation
[3]Polyline Features - QGIS documentation
[4]Polygon Features - QGIS documentation
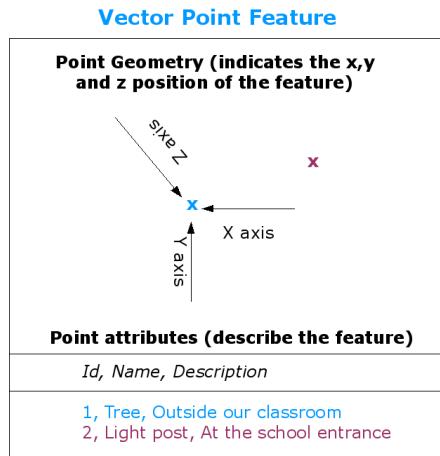
FIGURE 3.1: Point features
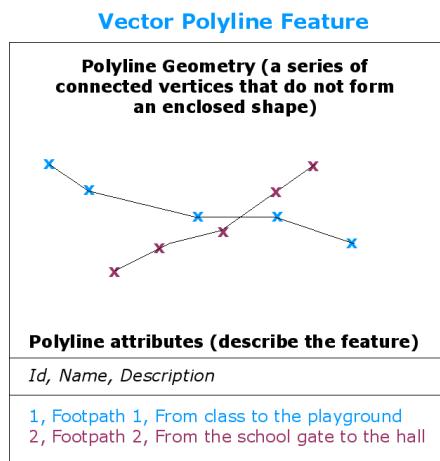
FIGURE 3.2: Polyline features

is often represented or encoded in this manner.

There are some downsides to working with vector data. It needs a large amount of ongoing work and long-term maintenance to preserve a high level of accuracy and reliability. Overshoots happen when line features (like roads) don't meet other line features *exactly* at an intersection. Undershoots happen when line features (like rivers) fall short of meeting other line features before the point where they should be connected. [5].

3.1.1 Shape-files

Shape-files are a very commonly used form of vector data for storing and representing the geographic location and attributes of geographic information. As the name suggests, shape-files are used to identify the "shape" of a region - aka the boundaries of the region being studied. Figure 3.4 illustrates what a shape-file for The Netherlands looks like.

---

[5]Common problems with vector data - QGIS documentation
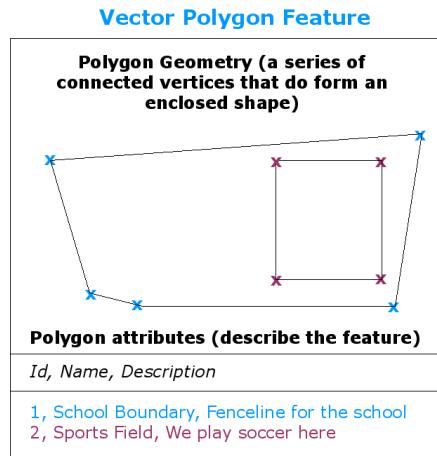
FIGURE 3.3: Polygon features

3.2 Raster data

Raster data is a matrix of pixels (called cells), wherein each pixel contains information specific to the area covered by that particular pixel/cell. Rasters can include categorical (e.g. "forest", "building", etc.) or continuous data (e.g. elevation). Certain features cannot be represented easily using vector data. Grasslands for example, would lose a lot of their contextual information (like variations in colour, plant type, and density of cover) if represented as a vector file. This happens because if you assign an attribute to a vector feature, it will be automatically applied to entire feature. Essentially, any feature that is **not** completely homogeneous cannot be accurately represented as vector data.[6]

"*Raster data is not only good for images that depict the real world surface (e.g. satellite images and aerial photographs), they are also good for representing more abstract ideas. For example, rasters can be used to show rainfall trends over an area, or to depict the fire risk on a landscape. In these kinds of applications, each cell in the raster represents a different value e.g. risk of fire on a scale of one to ten.*"[7]

3.2.1 Raster data in satellite images

In the context of satellite images, it's important to grasp the concept of **raster bands**. Using Landsat-8 images as our frame of reference, we see that every single image captured by Landsat-8 is made up of multiple layers, called "raster bands". The key word here is *bands*, and it refers to a portion of the electromagnetic spectrum (visible, infrared, and ultraviolet) that is captured by each of the sensors present on Landsat-8. The spectral bands along with their description and wavelength are provided in Table 3.1

3.3 Geo-referencing

Geo-referencing is the process of adding geographical information to a digital image, like a satellite photograph. This is done so that GIS software can accurately

---

[6]Raster data - QGIS documentation.
[7]Introduction to raster data - QGIS documentation.

FIGURE 3.4: Netherlands shape-file

Band	Description	Wavelength (μm)
B1	Coastal/aerosol	0.43-0.45
B2	Visible blue	0.45-0.51
B3	Visible green	0.53-0.59
B4	Visible red	0.64-0.67
B5	NIR	0.85-0.88
B6	SWIR 1	1.57-1.65
B7	SWIR 2	2.11-2.29
B8	Panchromatic	0.50-0.68
B9	Cirrus	1.36-1.38
B10	Thermal Infrared (TIRS) 1	10.6-11.19
B11	Thermal Infrared (TIRS) 2	11.50-12.51

TABLE 3.1: Landsat-8 spectral bands

pinpoint the "real world" location of the data present in a digital image. [8]

3.4 Ground Truth data

The "ground truth", as the name suggests, is the reality that we want to model or predict. Framed in the context of DL in remote sensing, "during inference, a classification model predicts a label, which can be compared with the ground truth label". In other words, the ground truth data tells us if our label prediction was right or not.

---

[8]Encyclopedia of Database Systems, pp 1246–1249

Chapter 4

Original approach/methodology

4.1 Study area

The original paper looked at 15 districts in the Khyber Pakhtunkhwa (KP) province in Pakistan. These districts are shown in Figure 4.1. Their names and corresponding coordinates are listed in Table 4.1.
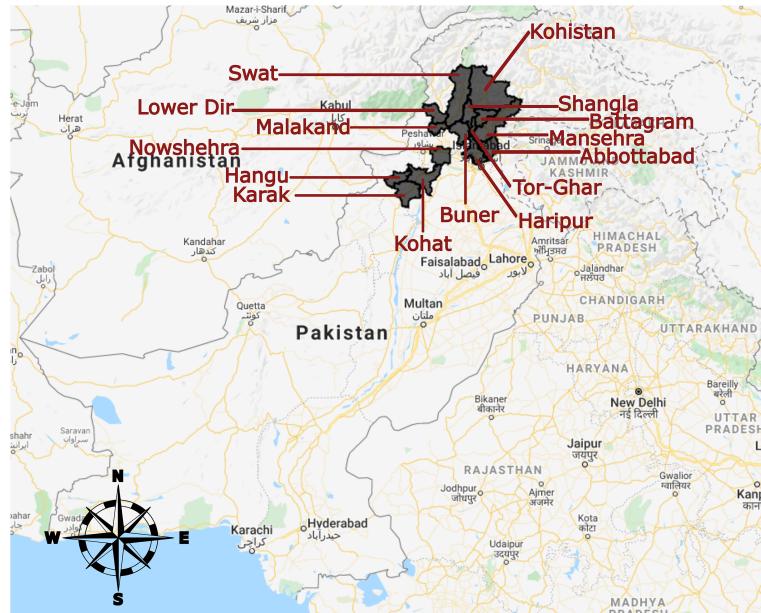
FIGURE 4.1: Geographic location of the districts under study

4.2 AI-ForestWatch framework

Figure 4.2 is taken from the original paper [103] and it represents the workflow as carried out by the authors. We begin with a brief step-by-step description of the original approach that was executed for the original study area which is then further expounded upon in ensuing subsections. We must note at this point that our replication of AI-ForestWatch differs from the original approach with regard to the generation of ground truth data. This will be detailed in later sections.

1. "Clean" Landsat-8 images are created by selecting only those that have less than 10% cloud cover.

BTAP Locations	Coordinates (longitude, latitude)
Hangu	(70.49 deg, 33.21 deg), (71.24 deg, 33.61 deg)
Karak	(70.75 deg, 32.79 deg), (71.49 deg, 33.38 deg)
Kohat	(71.05 deg, 33.75 deg), (72.03 deg, 33.05 deg)
Nowshehra	(71.68 deg, 34.14 deg), (72.26 deg, 33.69 deg)
Battagram	(72.85 deg, 34.97 deg), (73.51 deg, 34.55 deg)
Abbottabad	(72.96 deg, 34.37 deg), (73.52 deg, 33.81 deg)
Kohistan	(72.68 deg, 35.89 deg), (73.97 deg, 34.89 deg)
Haripur	(72.47 deg, 34.45 deg), (73.27 deg, 33.75 deg)
Tor-Ghar	(72.71 deg, 34.78 deg), (72.94 deg, 34.33 deg)
Mansehra	(72.81 deg, 35.18 deg), (74.17 deg, 34.20 deg)
Buner	(72.20 deg, 34.72 deg), (72.78 deg, 34.14 deg)
Lower Dir	(71.50 deg, 35.07 deg), (72.20 deg, 34.62 deg)
Malakand	(71.63 deg, 34.66 deg), (72.24 deg, 34.36 deg)
Shangla	(72.52 deg, 35.17 deg), (73.02 deg, 34.52 deg)
Swat	(72.08 deg, 35.89 deg), (72.86 deg, 34.56 deg)

TABLE 4.1: Coordinates of the original study area

2. The PDF land use maps for each district are geo-referenced and digitised, creating the ground truth data of red, green, and black pixels.

3. The UNet is trained using a combination of the Landsat-8 image and the ground truth.

4. After training, inference is performed on the images from 2014 to 2020, which yields a series of forest cover maps for each district/year combo.

5. Finally, a pixel-wise difference between the forest cover maps is calculated to generate the change maps. These maps indicate the change in forest cover for a particular district.

4.2.1 Dataset preparation

Satellite images are available to the public from a variety of different remote sensing programs such as Landsat, MODIS, and Sentinel. AI-ForestWatch uses data from Landsat-8, which was launched in 2013 with a temporal resolution of 16 days and a spatial resolution of 30 metres per pixel.[73]. The authors of AI-ForestWatch used Landsat-8 "Top of Atmosphere" (or TOA) imagery, which has data available from 2013 onward.
Using the Earth Engine JavaScript API[1], images that had less than 10% cloud cover (for each district/year) were selected. Then the set of these images was sent through a median filter that determined the median value of each pixel in each raster band of the Landsat-8 image and generated a clean composite image of the

---
[1]Get Started with Earth Engine
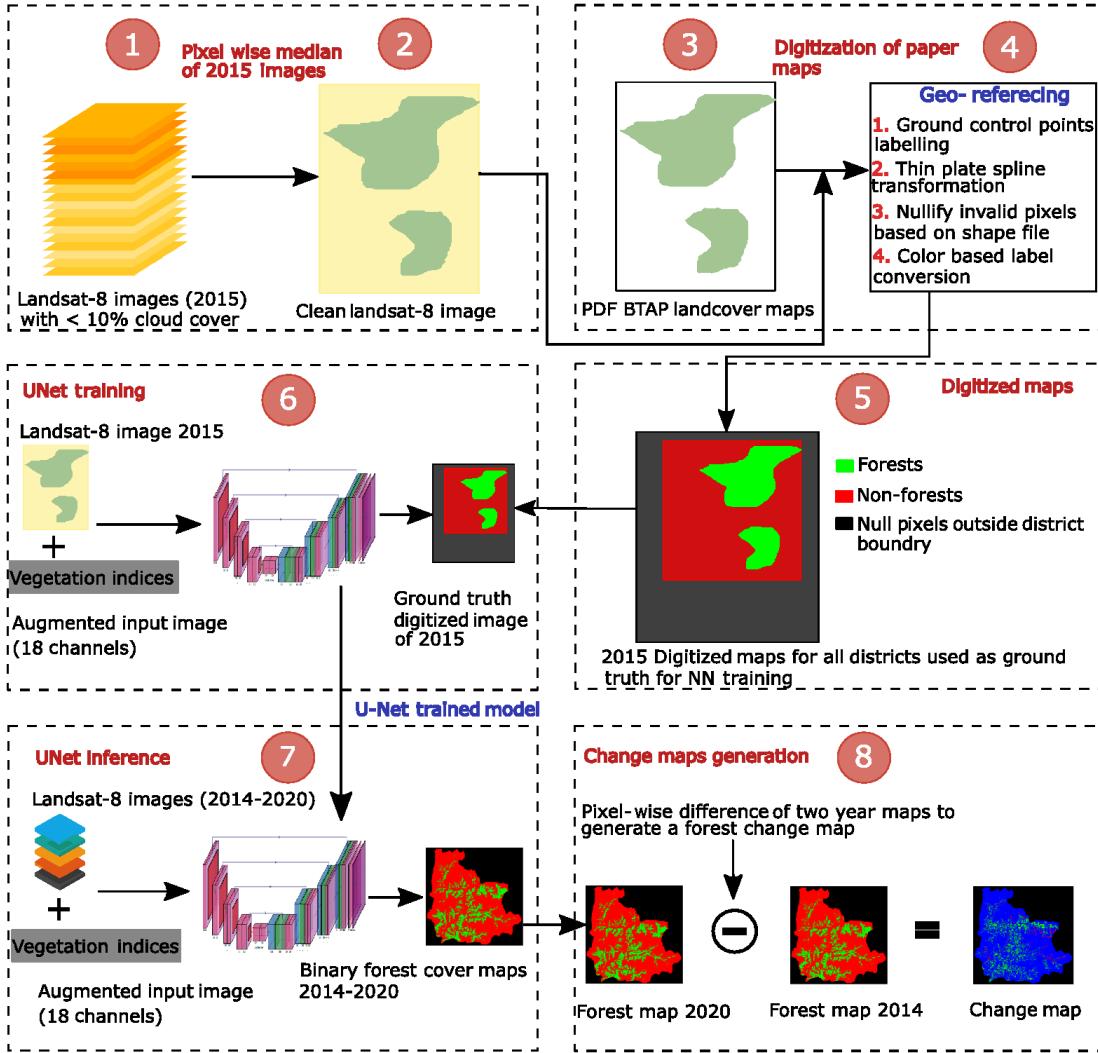
FIGURE 4.2: AI-ForestWatch framework

land cover for that region for 1 year. The resulting image served as the input to the AI-ForestWatch framework.

As outlined in Section 3.2.1, Landsat-8 captures data across 11 spectral bands (listed in Table 3.1) at a spatial resolution of 30 metres per pixel. Considering the high resolution of these images, 7 additional vegetation indices are appended to the image as 7 extra bands. These indices provide highly useful information for land cover classification.[100] Their definitions and corresponding formulae are given below. [96]

1. Normalised difference vegetation index (NDVI) - Chlorophyll absorbs the visible red wavelength and reflects infrared wavelength, which is the property used to calculate NDVI. The value is always between -1 and $+1$. A value close to -1 indicates water, a value close to 0 indicates barren land and a value close to $+1$ means healthy vegetation.[2]

$$\text{NDVI} = B5 - B4 \ / \ B5 + B4 \tag{4.1}$$

---

[2]NDVI from USGS

2. Enhanced vegetation index (EVI) - Similar to NDVI and can be used to quantify vegetation greenness. However, EVI corrects for certain atmospheric conditions and canopy background noise. It is also more sensitive in areas with dense vegetation.[3]

$$\text{EVI} = 2.5 * ((B5 - B4) \, / \, (B5 + 6 * B4 - 7.5 * B2 + 1)) \tag{4.2}$$

3. Soil adjusted vegetation index (SAVI) - Used to account for the effect of soil brightness in areas with low vegetation cover.[4]

$$\text{SAVI} = 1.5 * ((B5 - B4) \, / \, (B5 + B4 + 0.5)) \tag{4.3}$$

4. Modified soil adjusted vegetation index (MSAVI) - Helps mitigate the effect of soil brightness on the SAVI.[5]

$$\text{MSAVI} = 0.5 * (2 * B5 + 1 - \sqrt{(2 * B5 + 1)^2 - 8 * (B5 - B4))} \tag{4.4}$$

5. Normalised difference moisture index (NDMI) - "Used to determine vegetation water content".[6] Computed as a ratio of NIR and SWIR in traditional fashion.

$$\text{NDMI} = B5 - B6 \, / \, B5 + B6 \tag{4.5}$$

6. Normalised burn ratio (NBR) - Used to detect the existence and severity of burned regions.[7]

$$\text{NBR} = B5 - B7 \, / \, B5 + B7 \tag{4.6}$$

7. Normalised burn ratio-2 (NBR2) - Alters NBR to accentuate the water sensitivity in vegetation and could potentially be useful in post-fire recovery research.[8]

$$\text{NBR2} = B6 - B7 \, / \, B6 + B7 \tag{4.7}$$

4.2.2   Input file configurations

Four different configurations of the input files were tested. The only difference between these configurations was the number of input channels i.e., bands. These are as follows:

- **RGB** - Only 3 bands (B2, B3, and B4) from the Landsat-8 images are given as input to the model, in which case the input image size to the network is 128 × 128 × 3.

- **Full spectrum** - All 11 Landsat-8 bands listed in Table 3.1 are given as input to the model, in which case the input image size is 128 × 128 × 11.

- **Vegetation indices** - The 7 indices described earlier are given as input to the model, in which case the input image size is 128 × 128 × 7.

---

[3]EVI from USGS
[4]SAVI from USGS
[5]MSAVI from USGS
[6]NDMI from USGS
[7]NBR from USGS
[8]NBR2 from USGS

- **Augmented** - The 7 indices are stacked with the 11 Landsat-8 bands to create an 18 channel input, in which case the input image size is 128 × 128 × 18.

4.2.3   Generating ground truth data

As outlined in Section 3.4, the ground truth data lets us know the reality on the ground. The authors carried out a series of steps to generate ground truth data based on the land cover maps for each district made publicly available by the provincial government, for the year 2015 only. An open source tool named QGIS was utilised for the digitisation task. It has a built-in Georeferencer which simplifies this process.[9]

1. The land cover maps were used as a reference for digitisation. A sample map for Abbottabad is depicted in Figure 4.3. As we can see, there are 10 classes being labelled. Since we only detect forests, the remaining 9 classes are combined to form a single non-forest class.

2. Distinct features (like lakes and rivers) present in the Landsat-8 images and the land cover maps are marked as our Ground Control Points (GCP). This step correlates points on a satellite image with their actual locations on the land cover map.

3. Thin plate spline was the transformation utilised. The land cover map was placed on top of the corresponding Landsat-8 image to ensure that the GCP's lined up exactly on both images. This created a digitised map. According to the authors of AI-ForestWatch, "*For all the maps digitised, the georeferencing error was <1 × 10^{-13}.*"[103]

4. Since the land cover maps had already colour-coded forest areas as dark green, all dark green pixels in the digitised map were converted to "*forest*" labels and the rest were converted to "*non-forest*" labels.

5. Finally, individual shape-files for each district were used to mark all pixels outside the boundaries of that district as invalid/NULL, since the labelling only occurs within the borders of a particular district.

The results at different steps of the above-mentioned process are shown in Figure 4.4. These digitized maps are in the form of binary images with each pixel labelled as 2 or 1 indicating forest/non-forest for each pixel while 0 indicates an invalid/NULL pixel. NULL pixels are used for training but test scores are reported only for the forest/non-forest classes as a binary classification task. A similar procedure is adopted for all 15 districts. These digital maps are used as target annotations for the rest of our research. It is important to mention here that no numerical assessment of the accuracy of these maps was performed as the available ground truth data (i.e., the land cover maps) were paper maps.

4.2.4   Forest estimation using semantic segmentation

The AI-ForestWatch framework has a 2-stage forest change detection pipeline. First, pixel-wise segmentation is performed on a trained UNet (Step 6 in Figure 4.2) to generate binary forest/non-forest cover maps (Step 7 in Figure 4.2). Second,

---

[9]QGIS - Georeferencer tutorial
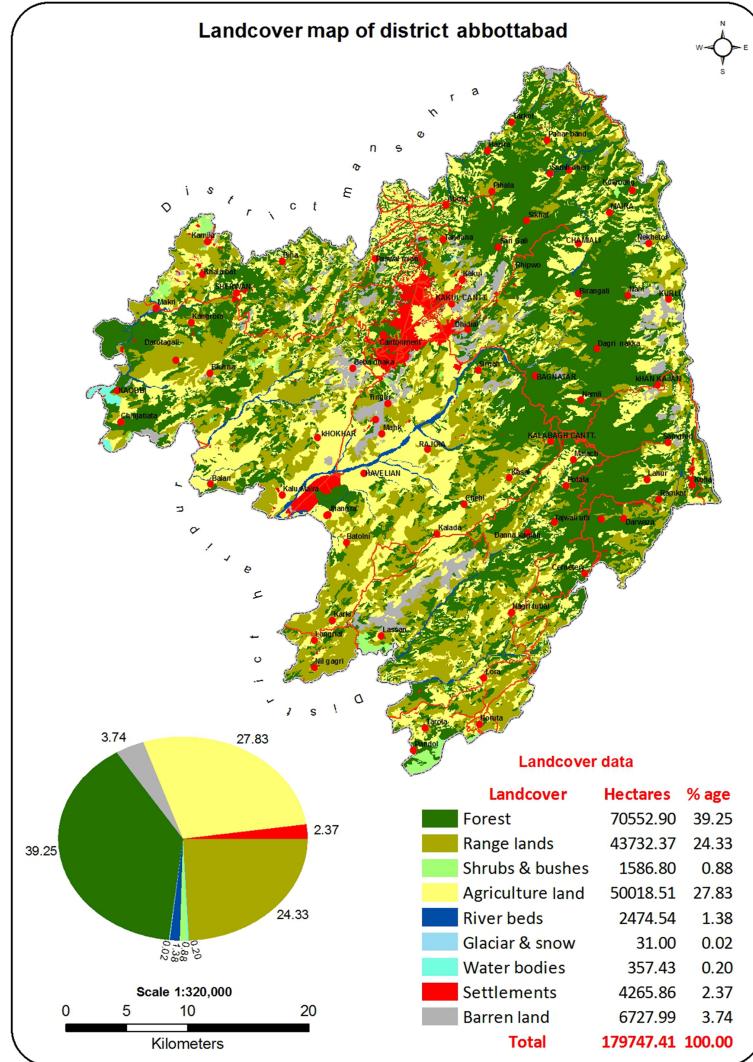
FIGURE 4.3: Abbottabad land cover map of 2015

pixel-wise difference between two consecutive binary forest/non-forest cover maps generates a change map (Step 7 in Figure 4.2). For pixel-wise segmentation, a UNet topology as shown in Figure 4.5 with encoder and decoder[78] stages is used. The operators used in the encoder are convolution, batch normalisation, activation, and pooling. Four such modules comprise the encoder section of the UNet topology, and hence it's called a four-stage encoder. The other part of the UNet is the decoder. It is the mirror image of the first half, meaning it is a four-stage decoder. Each module in a decoder consists of transposed convolution operation followed by a copy-and-fuse connection between the corresponding stages of encoder and decoder. It is followed by convolution, batch normalization, activation, and pooling layers. This concatenation operation of encoder and decoder outputs allows the decoder to utilise the features extracted by the encoder at subsequent stages. The detailed network topology is shown in Figure 4.5. The encoder (dotted square on the left) is where the input tensor is down-sampled and encoded into a smaller dimensional vector. The decoder (dotted square on the right) decodes this vector and produces full resolution segmentation for the input image. Each decoder module concatenates its output with the encoder output at the
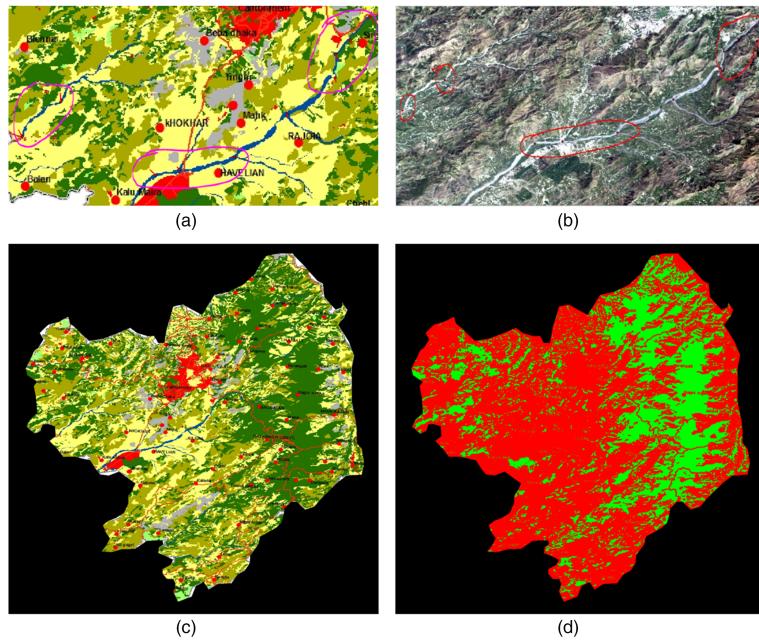
FIGURE 4.4: The digitisation steps for BTAP land cover maps into binary forest cover maps. (a) A portion of the ground truth land cover map of district Abbottabad. GCPs being labelled are encircled in pink colour. (b) The same portion of district Abbottabad as it appears in its corresponding Landsat-8 image of 2015. Same GCPs are encircled in red colour for comparison. (c) The land cover map of Abbottabad after georeferencing the map onto the Landsat-8 image of 2015. It has also been processed using its shape file to nullify all pixels outside the district boundary in order to extract only valid pixels for training. The dark green pixels are colour coded in the ground truth map indicating forest pixels. (d) Final digitized map of district Abbottabad after converting all dark green pixels to forest label, shown in green, and all the other non-forest classes into non-forest label, shown in red. The NULL pixels are shown in black which includes all pixels outside the district boundary.

corresponding stage. The final decoder tensor is passed through a convolution layer again followed by a softmax layer, which normalises the output probabilities for both classes in each pixel.

Implementation and training

The deep learning part of the AI-ForestWatch framework is implemented in Python using the PyTorch framework. The district images (at 30m per pixel spatial resolution) were quite large in volume so the authors divided them into smaller patches of size $256 \times 256 \times C$ pixels, where C is the number of channels in the input image - 3 for RGB, 11 for full spectrum, 7 for vegetation indices, and 18 for augmented input image. All district images are of different sizes, and with an average image size of $4000 \times 4000 \times C$ pixels, a dataset of 3375 $[(4000/256) * (4000/256) * 15]$ image patches for training and testing are created. 80% of these patches were randomly chosen for training, 10% for validation and 10% for testing. Table 4.2 presents the number of patches used for each of the aforementioned sets. The random selection of patches for each set resulted in
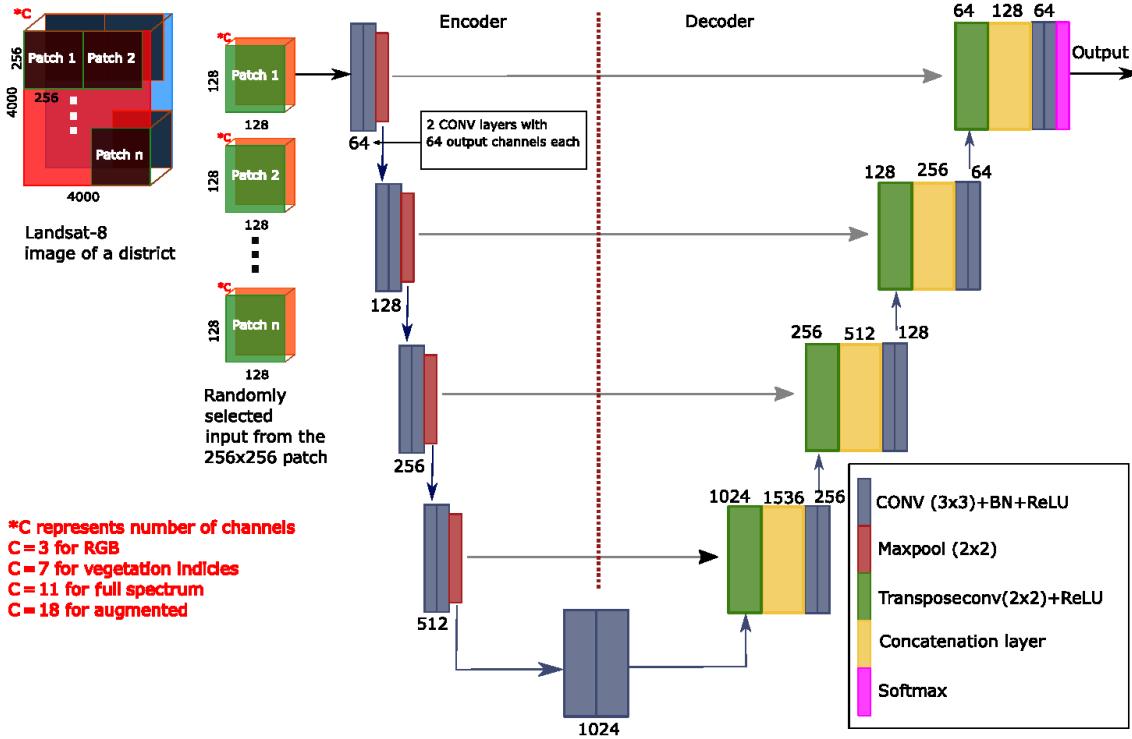
FIGURE 4.5: UNet topology for the AI-ForestWatch framework to detect the forest change. The arrows from encoder toward decoder indicate tensor outputs (feature maps) from the encoder being copied and concatenated with the transposed convolution decoder output along the depth dimension.

unique parts of each district going into the training, validation, and test sets, ensuring that maximum distribution was captured. The network itself accepts inputs of size $128 \times 128 \times C$, which are randomly cropped from these patches at training time (Step 6 in Figure 4.2).

Furthermore, for fast training of the model, the encoder layers are initialized by pre-trained weights of the VGG layers[84]. At inference, these patches are generated directly with dimensions of $128 \times 128 \times C$ in order to cover the whole image in a raster scan manner. The network was trained using back propagation with the RMSprop optimiser. The learning rate was scheduled to drop exponentially from an initial value of 1×10^{-06}. The model was trained for 200 epochs with a batch size of 64. The loss gradients were clipped at 0.05 to avoid exploding gradients. The training and testing was performed only for 2015 since that was the sole year for which annotated maps were available. The authors also experimented with different loss functions including cross-entropy loss function, focal loss[60] and dice loss[88] in both weighted and unweighted settings. Unweighted focal loss outperformed the other losses and hence it was adopted for this work.

4.2.5 Change detection statistics

Since the change is detected between consecutive years, the metrics devised in the original paper are also based on two images under inspection. These metrics were combined temporally to assess overall change trends. When 2 consecutive forest

Band	Percentage	Number of images
Training	80	2700
Validation	10	338
Test	10	337
Overall	100	3375

TABLE 4.2: Number of patches in the train, validation, and test sets

cover maps generated by the classification model are subtracted, a pixel-wise change map is obtained. Based on this the net change, gain, and loss in our regions of interest can be calculated. These metrics are explained as follows.

1. Forest cover percentage - This is calculated on each individual forest cover map generated by the model. It is the ratio of the number of forest pixels to the sum of forest and non-forest pixels in a classified image. It provides a percentage of the forest cover in the district under inspection but no information about change.

$$FC\% = \text{forest pixels} / \text{forest pixels} + \text{non-forest pixels} \qquad (4.8)$$

2. Gain percentage - Computed on the change map and it is the ratio of the gain pixels (-1) to the total number of change map pixels (-1, 0, and 1)

$$GP\% = \text{gain pixels} / \text{gain pixels} + \text{loss pixels} + \text{no change pixels} \qquad (4.9)$$

3. Loss percentage - Computed on the change map and it is the ratio of the loss pixels (+1) to the total number of change map pixels (-1, 0, and 1)

$$LP\% = \text{loss pixels} / \text{gain pixels} + \text{loss pixels} + \text{no change pixels} \qquad (4.10)$$

4. Effective forest cover change percentage - Effective forest cover change percentage (ECP%) is also calculated on the change map in order to compute the overall change in forest percentage with respect to the former year's forest percentage.

$$ECP\% = \text{forest\% latter year} - \text{forest\% former year pixels} / \text{forest\% former year} \qquad (4.11)$$

Chapter 5

Reproducing AI-ForestWatch

As stated earlier in Section 1.1.1, the goal of a direct (or one-to-one) reproduction is to successfully re-create results exactly as they are presented by the authors of the original publication, or at least results that are very very close to the original. With software, a little bit of variation is to be expected. We started by creating a fork of the original repository[1] on our own GitHub profile[2]. For the purposes of actually executing the code, we relied on a Linux-based high performance compute cluster provided by the university called DelftBlue[1], often shortened to DHPC.[3]

The authors tested 4 different configurations of the input file, as outlined in Section 4.2.2. For each of these configurations, along with testing the AI-ForestWatch framework itself, 3 other ML algorithms were explored - *decision tree, logistic regression, and random forest*. 5000 training points and 5000 testing points were chosen for each classifier. Landsat-8 images were used as input while the digitised data was used as ground truth.

The results of exploring different input configurations on all 4 algorithms are presented in Tables 5.1 to 5.4. The semantic segmentation-based approach typically outperformed the ML algorithms in terms of accuracy. Due to the inherent class imbalance between the forest and non-forest classes, a combination of F1 score, precision, recall, and accuracy was used to evaluate the performance of the models.

The key takeaways are:

- The RGB model yielded the best accuracy for semantic segmentation, but it lacked in terms of the F1 score and recall for the forest class.

- The vegetation indices model gave the best results in terms of F1 score and recall for the forest class but unfortunately lagged behind in precision.

- The augmented model with 18 input channels (*11 Landsat-8 bands + 7 vegetation indices*) produced the best results with regard to all measures except recall for the forest class.

Since the 18-channel input provided the best **overall** results, the authors chose the augmented model to perform forest estimation and change detection. The results that are presented from here onward were derived using this input configuration.

---

[1]Original repo
[2]Forked repo
[3]Welcome to DHPC.

Classifier	Precision		Recall		F1 score		Test accuracy (%)
	Forest	**Non-forest**	**Forest**	**Non-forest**	**Forest**	**Non-forest**	
Decision tree	0.41	0.82	0.50	0.75	0.45	0.78	68.78
Logistic regression	**0.75**	0.75	0.05	**0.99**	0.09	0.86	75.45
Random forest	0.57	0.84	**0.50**	0.87	**0.53**	0.85	77.60
Semantic segmentation (UNet)	0.69	**0.86**	0.41	0.95	0.52	**0.90**	**83.88**

TABLE 5.1: ML vs. UNet - RGB configuration

Classifier	Precision		Recall		F1 score		Test accuracy (%)
	Forest	**Non-forest**	**Forest**	**Non-forest**	**Forest**	**Non-forest**	
Decision tree	0.45	0.84	0.56	0.77	0.50	0.80	71.31
Logistic regression	**0.75**	0.75	0.05	**0.99**	0.09	0.86	75.45
Random forest	0.65	**0.87**	**0.60**	0.89	**0.62**	0.88	81.67
Semantic segmentation (UNet)	0.62	0.86	0.43	0.93	0.51	**0.89**	**82.52**

TABLE 5.2: ML vs. UNet - Full spectrum configuration

Classifier	Precision		Recall		F1 score		Test accuracy (%)
	Forest	**Non-forest**	**Forest**	**Non-forest**	**Forest**	**Non-forest**	
Decision tree	0.43	0.82	0.52	0.77	0.47	0.80	70.58
Logistic regression	**0.68**	0.82	0.39	**0.94**	0.50	0.87	79.89
Random forest	0.62	0.85	**0.53**	0.89	**0.57**	0.87	79.74
Semantic segmentation (UNet)	0.64	**0.87**	0.49	0.93	0.55	**0.90**	**83.45**

TABLE 5.3: ML vs. UNet - Vegetation indices configuration

Classifier	Precision		Recall		F1 score		Test accuracy (%)
	Forest	**Non-forest**	**Forest**	**Non-forest**	**Forest**	**Non-forest**	
Decision tree	0.45	0.84	0.56	0.77	0.50	0.80	71.45
Logistic regression	0.66	0.84	0.48	0.92	0.55	0.87	80.44
Random forest	0.65	0.87	**0.60**	0.89	**0.62**	0.88	81.74
Semantic segmentation (UNet)	**0.66**	**0.87**	0.48	**0.93**	0.55	**0.90**	**83.79**

TABLE 5.4: ML vs. UNet - Augmented configuration

5.1 Results and issues

After selecting the augmented model, forest cover and change maps for every year from 2014 to 2020 for all 15 district were generated. Since the model was trained and tested only on 2015 data, inference was performed on clean images for years 2014 and 2016 to 2020 using the augmented input configuration. The results presented in the original paper are categorised as follows:

1. Forest *cover* maps for each district/year combo;

2. Forest *change* maps for each district; and

3. Percentage of forest gain or loss

We faced a couple of small programming issues while reproducing the paper that were fixed quite easily.

- Permission issues when we attempted to run the code on Windows. Moving to the Linux-based DHPC solved it.

- Missing positional argument while running inference.

5.1.1 Incomplete input data

There was also trouble with the actual input data itself. The original paper states that "*Chitral and Upper-Dir districts were also not considered for this study, since most parts of these districts are covered with snow and we were unable to label any useful forest data points in this region.*"[103]. However, the inference data provided contains satellite images for both these districts. We skipped downloading the images for these 2 districts. The other issue was that they also provided the images for 2013 despite the fact that the paper only analyses data from 2014 to 2020, so we skipped the 2013 images too, in order to be as close to the original paper as possible. Lastly, a few images were missing from the Google Drive shared by the original authors on the official GitHub repository. For instance, images for the districts of Kohat and Kohistan for the year 2017 were missing, as were images for Buner and Lower-Dir for the year 2018. When we reached out to one of the authors, they gave us access to the code they executed on Google Earth Engine for downloading these images. We modified the district name and year, re-ran the code, and downloaded the missing images for ourselves.

Unfortunately we were only successful in reproducing the yearly forest *cover* maps. Our efforts to generate the forest *change* maps and statistics were futile. When we reached out to the authors regarding this, we were directed to one of their personal GitHub repositories containing scripts that were apparently used to generate the forest change maps and loss/gain stats.

We surmise that the project repository is newer by quite a length of time, as there is a very clear (temporal) gap between the official AI-ForestWatch repository and the authors' personal repository. One of the authors themselves emphasised that the code on their personal repository was old and wasn't shared with the public. Efforts to run these scripts and replicate the results were ultimately unsuccessful. The scripts didn't fail completely, but their output **did not resemble** what the original paper presented. The original results and our attempted reproduction are

shown in Figure 5.2 and Figure 5.1 respectively for comparison purposes. As can be seen, we were unable to generate the forest change map which would show the change in forest cover between 2014 and 2020.
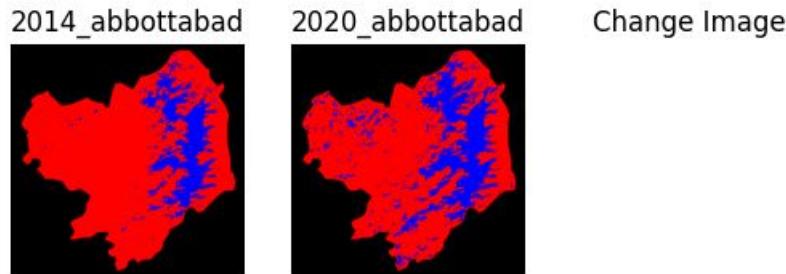
FIGURE 5.1: Our attempt at the forest change map
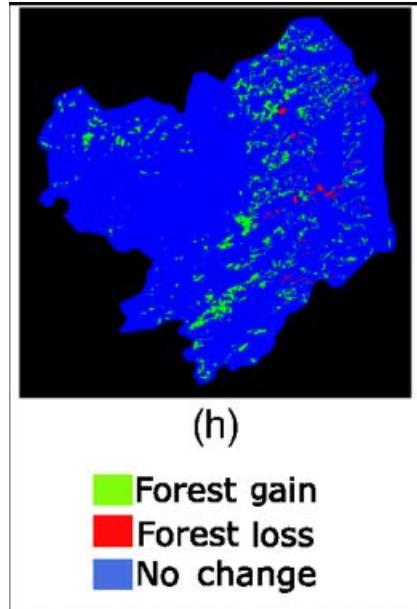
FIGURE 5.2: Original forest change map

5.1.2 Conclusions of the attempted reproduction

The yearly forest cover maps we generated seem to line up well with the samples presented in the original paper for Abbottabad and Battagram. The final resulting maps for the other 13 districts were not made publicly available by the authors.

Although we were not able to reproduce the forest change statistics, when we visually scrutinised the final PNG images created by running the inference code, they looked quite similar to the original results. The maps are too large to fit here, so they are made publicly available on our Google Drive.

For the purposes of this thesis, since we reproduced only 1 out of the 3 results that we aimed for (cover maps, change maps, and statistics), we deem this as constituting only a **partial reproduction** of AI-ForestWatch. It's not a complete failure, but the results aren't very encouraging either.

We speculate that the possible reason behind only a partial reproduction is that while the paper *explicitly* mentions generating forest change maps by performing "*multi-temporal change detection*", neither the code nor the underlying methodology were detailed. As noted earlier, the official repository on GitHub did not contain the code needed to generate either of those results. Although forest change maps and forest change statistics are presented in the original paper, most of the paper only describes pixel based classification of forest/non-forest data and generation of individual forest cover maps, which of course, we were able to successfully reproduce.

Another problem, as we've outlined in Section 5.1.1, was the incomplete input data, which meant that we had to download some of the images for a few districts ourselves. However, it seems very unlikely that this had a big impact, because most of the images we had to re-download were from several years ago (2017 & 2018). These were already archived by Google Earth Engine and thus would not have changed, i.e., the images would've been the same when the authors downloaded them. The results claimed in the original paper are detailed in Appendix A.

Chapter 6

Replicating AI-ForestWatch on The Netherlands

As stated earlier in Section 1.1.1, the goal of a replication is to see if we get similar results when we run the experiment on a different set of data. In this case, that would mean running AI-ForestWatch on a set of input files that it has never encountered before. To conduct a replication of AI-ForestWatch we used satellite images of The Netherlands, excluding the overseas territories. Similar to the original approach, we downloaded the images from Google Earth Engine. The decision was made to download individual files for each province from 2014 to 2021 owing to file size considerations. For example, one satellite image of the entire country for 2014 was 4GB, but downloading the data on a provincial basis resulted in a total file size of 2GB.

6.1 Ground truth data for The Netherlands

Section 4.2.3 details the process by which ground truth data for the study area were generated. This route was taken since reliable data for Pakistan was hard to come by. However, as mentioned earlier in Section 4.1, we took a different route. PDOK, an acronym for "*Publieke Dienstverlening Op de Kaart*" (Public Services On the Map), is an open-source platform created as a result of inter-agency cooperation within the Dutch Government. In collaboration with the "*Centraal Bureau voor de Statistiek*" (Central Agency for Statistics), they have published land use data for 2015[1] and 2017[2]. This data is in geopackage (or gpkg) format, so there was some pre-processing that had to be done before they could be used as ground truth data.

Land use in The Netherlands is officially classified under 37 different categories, all of which are listed in Table 6.1. We are concerned with number 23 on the list - "*bos*", aka the forest classification.

Although we now have a lot of land use data available, extracting only the forest labels is tricky, since a gpkg file is neither vector nor raster data. According to the OGC (Open Geospatial Consortium), "*a GeoPackage is the SQLite container and the GeoPackage Encoding Standard governs the rules and requirements of content stored in a GeoPackage container.*"[75] So we turned to the open source community for help in extracting what was needed. A question was asked on the GIS stack exchange forum, and it was answered in a few days, enabling us to create a proper ground

---

[1]Land use 2015
[2]Land use 2017

Categorie	English translation
Spoorterrein	Railroad
Hoofdweg	Highway
Vliegveld	Airport
Woongebied	Residential area
Detailhandel en horeca	Retail and catering
Openbare voorziening	Public facility
Sociaal-culturele voorziening	Socio-cultural facility
Bedrijfsterrein	Company premises
Stortplaats	Dump
Wrakkenopslagplaats	Wreckage Depot
Begraafplaats	Cemetery
Delfstofwinplaats	Mineral extraction site
Bouwterrein	Construction site
Semi-verhard overig terrein	Semi-paved other terrain
Park en plantsoen	Park and park
Sportterrein	Sports area
Volkstuin	Allotment garden
Dagrecreatief terrein	Day recreation area
Verblijfsrecreatief terrein	Recreational area
Glastuinbouw	Greenhouse horticulture
Overig agrarisch terrein	Other agricultural land
Bos	**Forest**
Open droog natuurlijk terrein	Open dry natural terrain
Open nat natuurlijk terrein	Open wet natural terrain
IJsselmeer & Markermeer	IJsselmeer & Markermeer
Afgesloten zeearm	Closed estuary
Rijn & Maas	Rhine & Meuse
Randmeer	Randmeer
Spaarbekken	Savings basin
Water met recreatieve functie	Water with recreational function
Water met delfstofwinningsfunctie	Water with mineral extraction function
Vloei- en/of slibveld	Liquid and/or sludge field
Overig binnenwater	Other inland waters
Waddenzee, Eems & Dollard	Wadden Sea, Ems & Dollard
Oosterschelde	Oosterschelde
Westerschelde	Western Scheldt
Noordzee	North Sea

TABLE 6.1: Land use categories for The Netherlands

truth data. The question and answer are linked here for reference.

The steps needed to extract the information we need from a *gpkg* file and convert that into a *GeoTIFF* file are given below:

1. Add a new column to the original land use layer.

2. Use the field calculator to set the value of the new column to 1. This will apply to all records/polygons.

3. Then use the select by attributes tool to select just the forest polygons,

4. Use the field calculator to set the value of the new column to 2. This will apply to just the forest records.

5. Rasterize the layer using the new column as the attribute to burn in

The field calculator mentioned in Step 2 is an option available in the open source QGIS tool, the tutorial for which is available on the QGIS website. After executing the steps listed above, we obtained ground truth data for The Netherlands for 2015. But since the land use gpkg file was not split by province, we added one final step of our own by splitting the final ground truth data into 12 different files - one for each province.

To accomplish this:

1. The ground truth data for the **entire country** was loaded into QGIS;

2. The shape-file for a province was placed on top of this image; and

3. The "**Clip Raster by Mask Layer**" option was used to extract data for that particular province

   - This option extracts only the data that lies **INSIDE** the boundary strictly defined by the shape-file overlaid in Step 2

The final clipping step was carried out for all 12 provinces.

Sample ground truth data for The Netherlands and for one province (Zuid Holland) for the year 2017 are shown in Figure 6.2 and Figure 6.1 respectively. The white pixels are forest labels (with a value of 2), grey pixels are non-forest labels (with a value of 1), and the rest are NULL pixels (with a value of 0). This numbering follows the convention set by the authors of AI-ForestWatch.

6.1.1   Pakistan vs. Netherlands - possible sticking points

The Pakistani landscape is an amalgamation of mountains, deserts, forests, and plateaus. In contrast, The Netherlands is almost entirely flat, except for some minor hilly areas in the South. The highest point in The Netherlands is 322.7 metres, whereas the highest point in Pakistan is literally the second tallest mountain in the world - K2, at a height of 8611 metres! However, the authors of AI-ForestWatch have avoided this by tackling it as a binary classification problem. Anything that wasn't explicitly tagged as "forest" in the land use map was blacked out i.e., tagged as "non-forest".
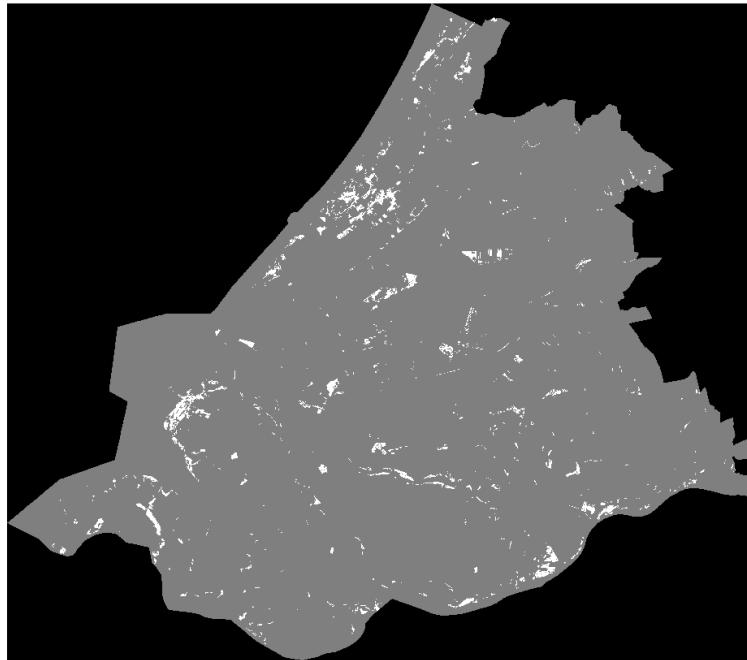
FIGURE 6.1: Zuid Holland 2017 ground truth data

Another potential hurdle was the fact that not all forests are created equal. There are different types with disparate characteristics and coverage on the earth's surface. This was overcome by relying on the fact that land use maps typically mark this information quite distinctly. For instance, the land cover map for Abbottabad shown in Figure 4.3 has 4 types of vegetation - "*forest*", "*range land*", "*shrubs and bushes*", and "*agriculture land*". The authors of AI-ForestWatch only considered areas that were explicitly tagged as "forest" on the land use map and everything else was considered "non-forest". Similarly, in the Netherlands land use categorisation shown in Table 6.1, "*bos*" is the explicit classification for "forest". There were others for agriculture and the like which we did not consider to be forest areas, and thus tagged them as "non-forest".

6.2 Results and issues

We ran into a few minor issues while replicating AI-ForestWatch on the NL data files:

- A portion of code that searched for input files based on a pre-determined naming scheme followed by the authors was modified to ensure that our Netherlands files were properly identified and picked up for processing.

- A missing function call while generating pickle files for training was fixed.

Similar to our experience while reproducing AI-ForestWatch (outlined in Chapter 5), we were only able to generate the yearly forest cover maps for The Netherlands using the 18-channel augmented input configuration. The files were too large to fit properly in this document, so they can be viewed on our Google Drive here. We did manage to generate the yearly forest cover maps for 2 other configurations (RGB
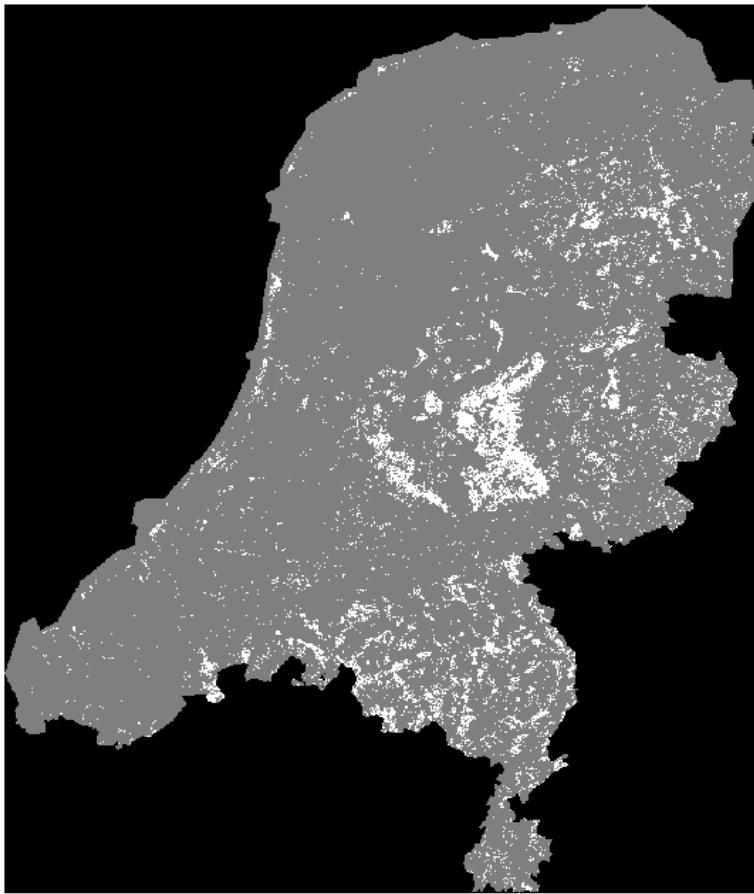
FIGURE 6.2: Netherlands 2017 ground truth data

and full spectrum), which is not something the original paper talks about or makes public. The authors merely noted that the augmented (18-channel) input configuration provided the best maps, and based on the files generated during the replication process, we agree.

6.2.1   Conclusions of the attempted replication

Similar to the experience we had while attempting a reproduction, we were able to replicate only 1 out of the 3 results that we aimed for (cover maps, change maps, and statistics). So we deem this as constituting only a **partial replication** of AI-ForestWatch. The provincial maps seem to be accurate upon a visual comparison with their corresponding ground truth data, but it is simply **not** consistently accurate year-over-year as we had hoped, such as Drenthe 2018 vs. Drenthe 2019 for example.

We surmise that a major reason for this is the limited availability of land use data. Even though The Netherlands had more high-quality data, it was still limited to just 2 years - 2015 and 2017. This could be overcome in the future as more land use data becomes available for other years. Additionally, even though we used an extra year of training data as compared to the original implementation, the effects of seasonality combined with median filtering may have contributed to the

irregularities seen in the final output.

Chapter 7

Final thoughts

At the start of this venture, we set out to find a project that captured our attention and that could also be replicated on a different dataset or use-case. After all, building upon the work done by other researchers is at the core of the scientific method. And so we were quite excited for the journey that lay ahead. What we found was that reproducibility in deep learning is not given the attention it deserves.

If you are thinking that it's difficult or perhaps even unfair to arrive at that conclusion after attempting a reproduction of just one paper, that would be a fair judgement. However our experience is not an isolated one. Based on existing literature that chronicles multiple attempts by others in this domain [43, 76, 3, 27, 26, 49, 67], one can conclude for themselves that ensuring replicability and reproducibility is clearly not a priority in the larger sphere of deep learning.

Of course, we did not expect this endeavour to be a breeze in any way. Replication studies are a complex undertaking and must not be taken lightly. Furthermore, even though deep learning is a vast and incredibly intricate field, it has iron-clad, verifiable mathematical foundations. We assumed that this would only help in our bid to replicate and reproduce a published DL paper/algorithm that caught our fancy. Unfortunately, this turned out to be a rather naive assumption. The whole exercise was unnecessarily complicated in our opinion. Deep learning algorithms are black boxes to begin with, so any additional complexity only serves to slow down the pace of innovation; moreover, it may also end up discouraging newcomers.

It is not all doom and gloom however. As the authors of AI-ForestWatch point out in their conclusion, this was a first attempt at assessing the real world effects of a public policy enacted by the local government of the KP province. And to their credit, the primary objective of generating yearly forest cover maps (of an under-studied region no less), was quite successful. We were able to reproduce the final forest cover maps on the **original input data** that was provided. A first time attempt like this lays the groundwork for improvements and refinements over time that make the algorithm more robust and more accurate in picking up changes in the input data.

Global forest cover information has been available for a while now, most notably in [83] and [33]. However, such global maps often lack localised/regional context. Forest species, coverage areas, quality, etc. are all highly dependent on local circumstances, and deep learning offers a viable way for us to take native data into

account while performing this type of analysis.

In the interest of furthering openness, we have made our fork of AI-ForestWatch available on our GitHub here.

7.1 Recommendations

We have devised a framework, not limited to any one programming languages, that we feel could assist future researchers (and grad students!) in improving the reproducibility and/or replicability of their work. This framework is outlined below:

1. Ensure that all data necessary to run your code is publicly available, preferably on a well known platform like Google Drive or Dropbox.

2. Ensure that all code is up-to-date and publicly available on a version control platform like GitHub.

    - Don't skip any specific code that helps generate a portion of the final results, no matter how simple it may seem.

3. Detail the exact reasons that prompted the choice of a particular algorithm or input configuration rather than a generic "this combination was best overall so we chose it".

Appendix A

Results presented in the original paper

Nowshehra was the only district to show a decrease in forest cover between 2014 and 2020. The rest of the districts show an increase. The **ECP%** column in Table A.1 is the percentage change in forest cover compared to 2014. This explains why certain changes are very high (+764% for Haripur and +327.13% for Malakand), since forest cover in these areas in 2014 was very low. A majority of the afforestation was executed in these districts. The '14 to '20 LP% and '14 to '20 GP% columns outline the forest loss and GP% respectively (as per the metrics outlined in Section 4.2.5). Except Nowshehra, all other districts show more gain% than loss%, thus explaining the overall gain in most districts.

Figure A.1 and Figure A.2 illustrate the forest cover and change maps as presented in the original paper.

Region	2014 FC%	2015 FC%	2016 FC%	2017 FC%	2018 FC%	2019 FC%	2020 FC%	'14 to '20 LP%	'14 to '20 GP%	'14 to '20 ECP%
Nowshehra	2.94	11.31	3.63	7.74	5.39	5.71	2.80	1.17	1.02	-4.92
Kohat	0.72	15.0	1.04	1.81	2.48	0.83	1.0	0.27	0.55	+38.93
Karak	1.08	13.84	2.85	5.21	3.45	3.56	2.51	0.35	1.78	+131.56
Hangu	0.15	4.03	1.62	4.53	1.62	1.76	0.58	0.11	0.54	+295.09
Battagram	17.84	23.92	23.96	25.90	27.81	28.18	33.80	1.92	17.88	+89.48
Abbottabad	17.22	29.33	15.84	24.95	30.68	17.16	23.53	0.72	7.03	+36.64
Haripur	0.75	21.78	1.27	12.40	17.14	5.00	6.50	0.21	5.87	+764.83
Kohistan	21.23	29.42	24.12	31.56	30.97	32.29	31.82	2.03	12.62	+49.88
Tor Ghar	15.95	25.43	16.89	24.55	35.67	22.04	25.76	0.48	10.29	+61.49
Mansehra	16.46	20.53	18.87	21.50	24.38	19.14	23.98	1.44	8.97	+45.72
Buner	2.97	25.63	5.63	21.40	18.50	8.83	12.19	0.24	9.46	+310.96
Lower Dir	4.29	11.94	10.54	12.53	11.88	4.43	7.71	1.06	4.49	+79.95
Malakand	0.94	13.15	11.01	15.11	8.63	6.62	4.03	0.24	3.33	+327.13
Shangla	17.68	31.91	27.81	32.02	31.45	25.30	32.17	0.83	15.32	+81.92

TABLE A.1: Forest stats presented in the original paper
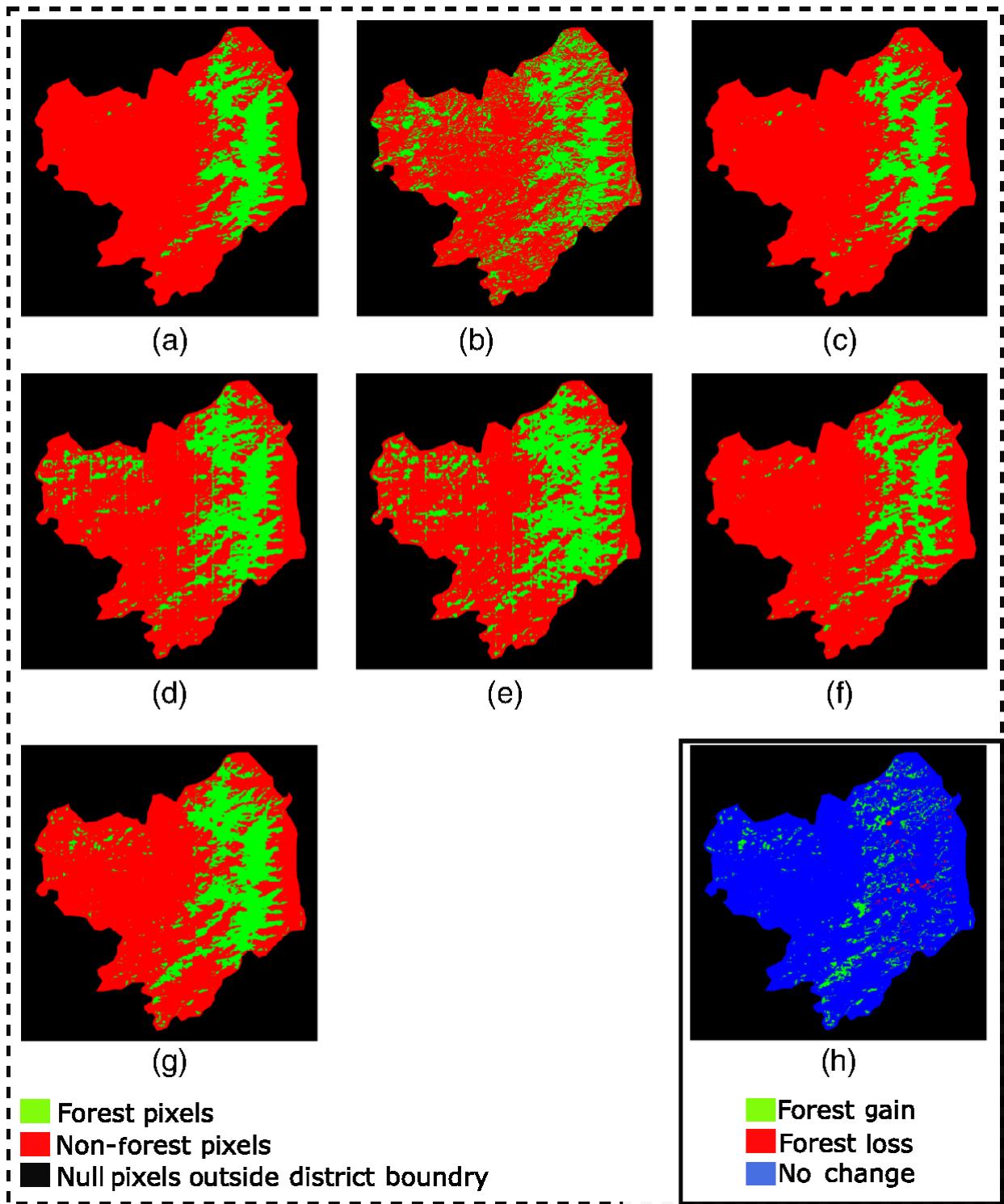
FIGURE A.1: Abbottabad district 2014 to 2020 statistics generated au-
tomatically by the AI-ForestWatch framework: (a)–(g) show the forest
cover maps generated by the UNet inference from years 2014 to 2020,
respectively, and (h) shows the forest change map of 2020 with re-
spect to 2014.
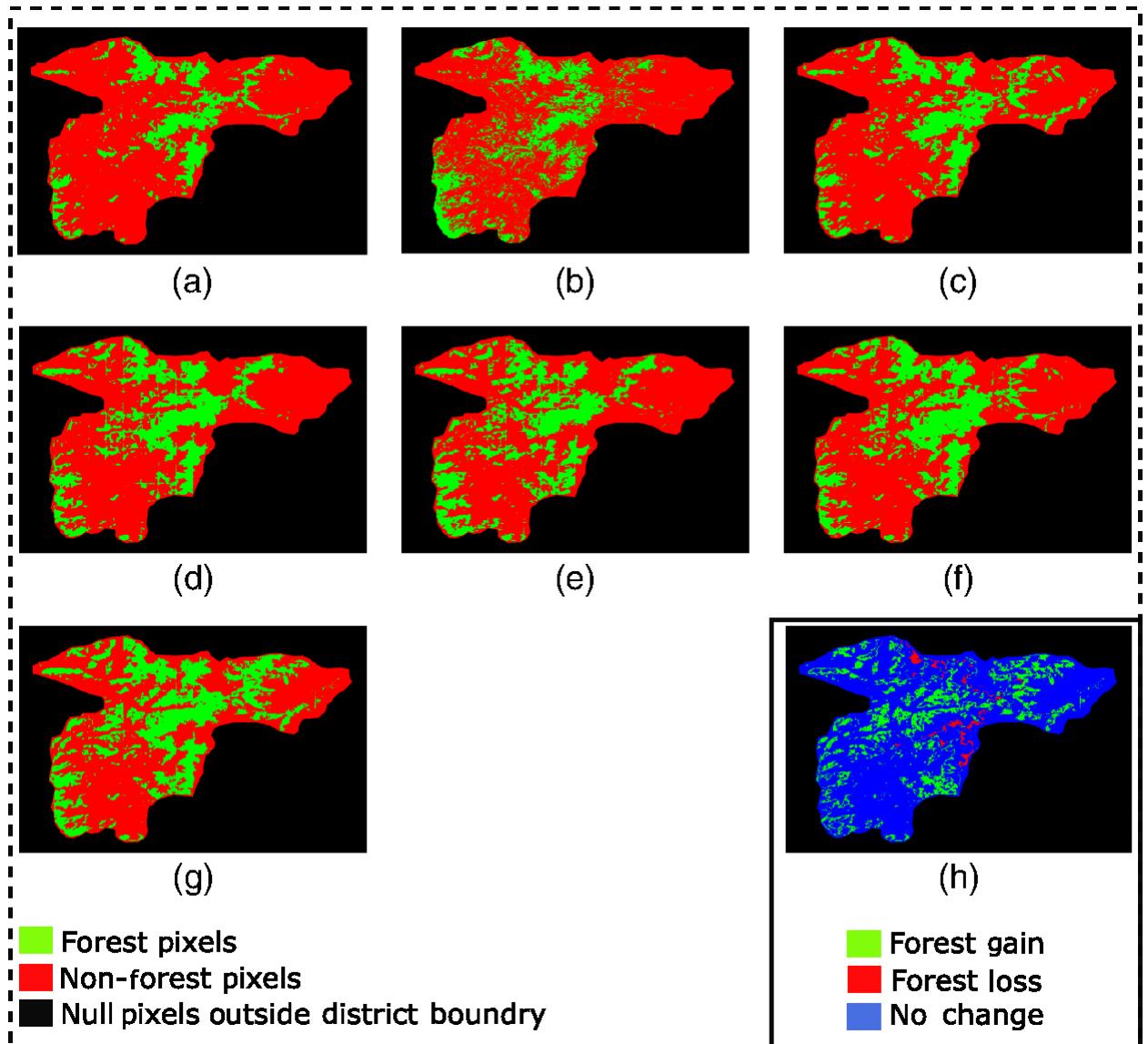
FIGURE A.2: Battagram district 2014 to 2020 statistics generated automatically by the AI-ForestWatch framework: (a)–(g) show the forest cover maps generated by the UNet inference from years 2014 to 2020, respectively, and (h) shows the forest change map of 2020 with respect to 2014.

Bibliography

[1]   Delft High Performance Computing Centre (DHPC). *DelftBlue Supercomputer (Phase 1)*. https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase1. 2022.

[2]   Vijay Badrinarayanan, Ankur Handa, and Roberto Cipolla. *SegNet: A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling*. 2015. URL: https://arxiv.org/abs/1505.07293.

[3]   M. Baker. "1,500 Scientists Lift the Lid on Reproducibility". In: *Nature* 533.7604 (2016), pp. 452–454.

[4]   Yifang Ban and Osama A. Yousif. "Multitemporal Spaceborne SAR Data for Urban Change Detection in China". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 5.4 (2012), pp. 1087–1094. DOI: 10.1109/JSTARS.2012.2201135.

[5]   T. Blaschke. "Object based image analysis for remote sensing". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 65.1 (2010), pp. 2–16. ISSN: 0924-2716. DOI: https://doi.org/10.1016/j.isprsjprs.2009.06.004. URL: https://www.sciencedirect.com/science/article/pii/S0924271609000884.

[6]   Mourad Bouziani, Kalifa Goita, and Dong-Chen He. "Automatic change detection of buildings in urban environment from very high spatial resolution images using existing geodatabase and prior knowledge". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 65.1 (2010), pp. 143–153.

[7]   Francesca Bovolo, Lorenzo Bruzzone, and Mattia Marconcini. "A novel approach to unsupervised change detection based on a semisupervised SVM and a similarity measure". In: *IEEE Transactions on Geoscience and Remote Sensing* 46.7 (2008), pp. 2070–2082.

[8]   C M R Caridade, A R S Marçal, and T Mendonça. "The use of texture for image classification of black & white air photographs". In: *Int. J. Remote Sens.* 29.2 (Jan. 2008), pp. 593–607.

[9]   Jonathan Cheung-Wai Chan, Kwok-Ping Chan, and Anthony Gar-On Yeh. "Detecting the nature of change in an urban environment: A comparison of machine learning algorithms". In: *Photogrammetric Engineering and Remote Sensing* 67.2 (2001), pp. 213–226.

[10]   Zhikang Chen, Christopher D Elvidge, and David P Groeneveld. "Vegetation change detection using high spectral resolution vegetation indices". In: *Remote sensing change detection: environmental monitoring methods and applications* (1998), pp. 181–190.

[11]   John B Collins and Curtis E Woodcock. "An assessment of several linear change detection techniques for mapping forest mortality using multitemporal Landsat TM data". In: *Remote sensing of Environment* 56.1 (1996), pp. 66–77.

[12] Pol R Coppin and Marvin E Bauer. "Digital change detection in forest ecosystems with remote sensing imagery". In: *Remote sensing reviews* 13.3-4 (1996), pp. 207–234.

[13] Eric P Crist. "A TM tasseled cap equivalent transformation for reflectance factor data". In: *Remote sensing of Environment* 17.3 (1985), pp. 301–306.

[14] X Long Dai and Slamak Khorram. "Remotely sensed change detection based on artificial neural networks". In: *Photogrammetric engineering and remote sensing* 65 (1999), pp. 1187–1194.

[15] Rodrigo Caye Daudt et al. "Urban Change Detection for Multispectral Earth Observation Using Convolutional Neural Networks". In: *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*. 2018, pp. 2115–2118. DOI: 10.1109/IGARSS.2018.8518015.

[16] Tim De Chant and Maggi Kelly. "Individual object change detection for monitoring the impact of a forest pathogen on a hardwood forest". en. In: *Photogramm. Eng. Remote Sensing* 75.8 (Aug. 2009), pp. 1005–1013.

[17] J.S. Deng et al. "PCA-based land-use change detection and analysis using multitemporal and multisensor satellite data". In: *International Journal of Remote Sensing* 29.16 (2008), pp. 4823–4838. URL: https://doi.org/10.1080/01431160801950162.

[18] Loïc Paul Dutrieux et al. "Monitoring forest cover loss using multiple data streams, a case study of a tropical dry forest in Bolivia". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 107 (2015), pp. 112–125. ISSN: 0924-2716. URL: https://www.sciencedirect.com/science/article/pii/S0924271615000970.

[19] Peter W Eklund, Jane You, and Peter Deer. "Mining remote sensing image data: an integration of fuzzy set theory and image understanding techniques for environmental change detection". In: *Data Mining and Knowledge Discovery: Theory, Tools, and Technology II*. Ed. by Belur V Dasarathy. Orlando, FL, Apr. 2000.

[20] Mulham Fawakherji et al. "Crop and Weeds Classification for Precision Agriculture Using Context-Independent Pixel-Wise Segmentation". In: *2019 Third IEEE International Conference on Robotic Computing (IRC)*. 2019, pp. 146–152. DOI: 10.1109/IRC.2019.00029.

[21] Peter Fisher et al. "Detecting change in vague interpretations of landscapes". en. In: *Ecol. Inform.* 1.2 (Apr. 2006), pp. 163–178.

[22] Peter F Fisher and Sumith Pathirana. "The ordering of multitemporal fuzzy land-cover information derived from landsat MSS data". In: *Geocarto Int.* 8.3 (Sept. 1993), pp. 5–14.

[23] G M Foody and D S Boyd. "Detection of partial land cover change associated with the migration of inter-class transitional zones". In: *Int. J. Remote Sens.* 20.14 (Jan. 1999), pp. 2723–2740.

[24] Jay Gao. *Digital analysis of remotely sensed imagery*. McGraw-Hill Education, 2009.

[25] Chandra Giri et al. "Status and distribution of mangrove forests of the world using earth observation satellite data". In: *Global Ecology and Biogeography* 20.1 (2011), pp. 154–159.

[26] Omar S Gómez, Natalia Juristo, and Sira Vegas. "Replications types in experimental disciplines". In: *Proceedings of the 2010 ACM-IEEE international symposium on empirical software engineering and measurement*. 2010, pp. 1–10.

[27] Omar S Gómez, Natalia Juristo, and Sira Vegas. "Understanding replication of experiments in software engineering: A classification". In: *Information and Software Technology* 56.8 (2014), pp. 1033–1048.

[28] Sucharita Gopal and Curtis Woodcock. "Remote sensing of forest change using artificial neural networks". In: *IEEE Transactions on Geoscience and Remote Sensing* 34.2 (1996), pp. 398–404.

[29] Patrick Griffiths et al. "Mapping megacity growth with multi-sensor data". en. In: *Remote Sens. Environ.* 114.2 (Feb. 2010), pp. 426–439.

[30] Ola Hall and Geoffrey J Hay. "A multiscale object-specific approach to digital change detection". en. In: *Int. J. Appl. Earth Obs. Geoinf.* 4.4 (Nov. 2003), pp. 311–327.

[31] Jiawei Han, Jian Pei, and Hanghang Tong. *Data mining: concepts and techniques*. Morgan kaufmann, 2022.

[32] Matthew C Hansen et al. "A method for integrating MODIS and Landsat data for systematic monitoring of forest cover and change in the Congo Basin". In: *Remote Sensing of Environment* 112.5 (2008), pp. 2495–2513.

[33] Matthew C Hansen et al. "High-resolution global maps of 21st-century forest cover change". In: *science* 342.6160 (2013), pp. 850–853.

[34] Robert M Haralick, K Shanmugam, and Its'hak Dinstein. "Textural Features for Image Classification". In: *IEEE Trans. Syst. Man Cybern.* SMC-3.6 (Nov. 1973), pp. 610–621.

[35] Geoffrey G Hazel. "Object-level processing of spectral imagery for detection of targets and changes using spatial-spectral-temporal techniques". In: *Algorithms for Multispectral, Hyperspectral, and Ultraspectral Imagery VII*. Orlando, FL: SPIE, Aug. 2001.

[36] Dong-Chen He and Li Wang. "Texture features based on texture spectrum". en. In: *Pattern Recognit.* 24.5 (Jan. 1991), pp. 391–399.

[37] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[38] Daniel Hölbling, Barbara Friedl, and Clemens Eisank. "An object-based approach for semi-automated landslide change detection and attribution of changes to landslide classes in northern Taiwan". In: *Earth Science Informatics* 8.2 (2015), pp. 327–335.

[39] Chengquan Huang et al. "Use of a dark object concept and support vector machines to automate forest cover change analysis". In: *Remote sensing of environment* 112.3 (2008), pp. 970–985.

[40] Masroor Hussain et al. "Change detection from remotely sensed images: From pixel-based to object-based approaches". In: *ISPRS Journal of photogrammetry and remote sensing* 80 (2013), pp. 91–106.

[41] Jungho Im and John R Jensen. "A change detection model based on neighborhood correlation image analysis and decision tree classification". In: *Remote Sensing of Environment* 99.3 (2005), pp. 326–340.

[42]   Jungho Im, John R. Jensen, and Michael E. Hodgson. "Optimizing the binary discriminant function in change detection applications". In: *Remote Sensing of Environment* 112.6 (2008), pp. 2761–2776. ISSN: 0034-4257. URL: https://www.sciencedirect.com/science/article/pii/S0034425708000400.

[43]   Darrel C. Ince, Leslie Hatton, and John Graham-Cumming. "The case for open computer programs". In: *Nature* 482.7386 (Feb. 2012), pp. 485–488. ISSN: 1476-4687. DOI: 10.1038/nature10836. URL: https://doi.org/10.1038/nature10836.

[44]   Kostiantyn Isaienkov et al. "Deep Learning for Regular Change Detection in Ukrainian Forest Ecosystem With Sentinel-2". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14 (2020), pp. 364–376.

[45]   John R Jensen et al. *Introductory digital image processing: a remote sensing perspective.* Ed. 2. Prentice-Hall Inc., 1996.

[46]   John R Jensen et al. *Introductory digital image processing: a remote sensing perspective.* Ed. 2. Prentice-Hall Inc., 1996.

[47]   L Jiang and R M Narayanan. "A shape-based approach to change detection of lakes using time series remote sensing images". In: *IEEE Transactions on Geoscience and Remote Sensing* 41.11 (2003), pp. 1035–1037.

[48]   David John and Ce Zhang. "An attention-based U-Net for detecting deforestation within satellite sensor imagery". In: *International Journal of Applied Earth Observation and Geoinformation* 107 (2022), p. 102685.

[49]   Natalia Juristo and Sira Vegas. "The role of non-exact replications in software engineering experiments". In: *Empirical Software Engineering* 16.3 (2011), pp. 295–324.

[50]   Richard J Kauth and GS Thomas. "The tasselled cap–a graphic description of the spectral-temporal development of agricultural crops as seen by Landsat". In: *LARS symposia*. 1976, p. 159.

[51]   Alphonse Kayiranga et al. "Monitoring forest cover change and fragmentation using remote sensing and landscape metrics in Nyungwe-Kibira park". In: *Journal of Geoscience and Environment Protection* 4.11 (2016), pp. 13–33.

[52]   Devendra Kurnar. "Monitoring forest cover changes using remote sensing and GIS: a global prospective". In: *Research Journal of Environmental Sciences* 5.2 (2011), p. 105.

[53]   Landsat Missions. *Imagery for Everyone - Timeline to Open Landsat Archive.* https://www.usgs.gov/media/files/imagery-everyone-timeline-open-landsat-archive. 2008.

[54]   Daniel T Larose and Chantal D Larose. *Discovering knowledge in data: an introduction to data mining.* Vol. 4. John Wiley & Sons, 2014.

[55]   Antoine Lefebvre, Thomas Corpetti, and Laurence Hubert-Moy. "Object-oriented approach and texture analysis for change detection in very high resolution images". In: *IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium*. Boston, MA, USA: IEEE, 2008.

[56]   Michael A Lefsky. "A global forest canopy height map from the Moderate Resolution Imaging Spectroradiometer and the Geoscience Laser Altimeter System". In: *Geophysical Research Letters* 37.15 (2010).

[57] Eric A. Lehmann et al. "Forest cover trends from time series Landsat data for the Australian continent". In: *International Journal of Applied Earth Observation and Geoinformation* 21 (2013), pp. 453–462. ISSN: 1569-8432. DOI: https://doi.org/10.1016/j.jag.2012.06.005.

[58] Eric A. Lehmann et al. "Updating the 2001 National Land Cover Database Impervious Surface Products to 2006 using Landsat Imagery Change Detection Methods". In: *Remote Sensing of Environment* 114.8 (2010), pp. 1676–1686. ISSN: 0034-4257. DOI: https://doi.org/10.1016/j.rse.2010.02.018.

[59] Thomas Lillesand, Ralph W Kiefer, and Jonathan Chipman. *Remote sensing and image interpretation*. John Wiley & Sons, 2015.

[60] Tsung-Yi Lin et al. "Focal loss for dense object detection". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2980–2988.

[61] Jia Liu et al. "A Deep Convolutional Coupling Network for Change Detection Based on Heterogeneous Optical and Radar Images". In: *IEEE Transactions on Neural Networks and Learning Systems* 29.3 (2018), pp. 545–559. DOI: 10.1109/TNNLS.2016.2636227.

[62] X. Liu and R. G. Lathrop Jr. "Urban change detection based on an artificial neural network". In: *International Journal of Remote Sensing* 23.12 (2002), pp. 2513–2518. URL: https://doi.org/10.1080/01431160110097240.

[63] Ivan Lizarazo and Joana Barros. "Fuzzy image segmentation for urban land-cover classification". en. In: *Photogramm. Eng. Remote Sensing* 76.2 (Feb. 2010), pp. 151–162.

[64] Dengsheng Lu et al. "Change detection techniques". In: *International journal of remote sensing* 25.12 (2004), pp. 2365–2401.

[65] Dengsheng Lu et al. "Land-cover binary change detection methods for use in the moist tropical region of the Amazon: a comparative study". In: *International journal of remote sensing* 26.1 (2005), pp. 101–114.

[66] RS Lunetta and CD Elvidge. "Applications, project formulation, and analytical approach". In: *Remote sensing change detection: Environmental monitoring methods and applications* (1998), pp. 1–19.

[67] Jonathan Lung et al. "On the difficulty of replicating human subjects studies in software engineering". In: *Proceedings of the 30th international conference on Software engineering*. 2008, pp. 191–200.

[68] William A Malila. "Change vector analysis: An approach for detecting forest changes with Landsat". In: *LARS symposia*. 1980, p. 385.

[69] Nikolay Malkin, Caleb Robinson, and Nebojsa Jojic. "High-resolution land cover change from low-resolution labels: Simple baselines for the 2021 IEEE GRSS Data Fusion Contest". In: *arXiv preprint arXiv:2101.01154* (2021).

[70] J-F Mas. "Monitoring land-cover changes: a comparison of change detection techniques". In: *International journal of remote sensing* 20.1 (1999), pp. 139–152.

[71] Graciela Metternicht. "Change detection assessment using fuzzy sets and remotely sensed data: an application of topographic map revision". en. In: *ISPRS J. Photogramm. Remote Sens.* 54.4 (Sept. 1999), pp. 221–233.

[72] Ofer Miller, Arie Pikaz, and Amir Averbuch. "Objects based change detection in a pair of gray-level images". en. In: *Pattern Recognit.* 38.11 (Nov. 2005), pp. 1976–1992.

[73]   NASA. *Landsat 8*. Dec. 2021. URL:
        https://landsat.gsfc.nasa.gov/satellites/landsat-8/.

[74]   I Niemeyer, P R Marpu, and S Nussbaum. "Change detection using object
        features". In: *Lecture Notes in Geoinformation and Cartography*. Lecture notes
        in geoinformation and cartography. Berlin, Heidelberg: Springer Berlin
        Heidelberg, 2008, pp. 185–201.

[75]   OGC. "OGC". In: *Open Geospatial Consortium*. 2022.

[76]   Roger D Peng. "Reproducible research in computational science". In: *Science*
        334.6060 (2011), pp. 1226–1227.

[77]   J A Richards. "Analysis of remotely sensed data: the formative decades and
        the future". In: *IEEE Trans. Geosci. Remote Sens.* 43.3 (Mar. 2005), pp. 422–432.

[78]   Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-Net:
        Convolutional Networks for Biomedical Image Segmentation". In: *Medical
        Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by
        Nassir Navab et al. Cham: Springer International Publishing, 2015,
        pp. 234–241. ISBN: 978-3-319-24574-4.

[79]   E Sali and H Wolfson. "Texture classification in aerial photographs and
        satellite data". en. In: *Int. J. Remote Sens.* 13.18 (Dec. 1992), pp. 3395–3408.

[80]   Robert A Schowengerdt. *Techniques for image processing and classifications in
        remote sensing*. Academic Press, 2012.

[81]   P Serra, X Pons, and D Sauri. "Post-classification change detection with data
        from different sensors: Some accuracy considerations". In: *Int. J. Remote Sens.*
        24.16 (Jan. 2003), pp. 3311–3340.

[82]   Masanobu Shimada et al. "New global forest/non-forest maps from ALOS
        PALSAR data (2007–2010)". In: *Remote Sensing of environment* 155 (2014),
        pp. 13–31.

[83]   Masanobu Shimada et al. "New global forest/non-forest maps from ALOS
        PALSAR data (2007–2010)". In: *Remote Sensing of environment* 155 (2014),
        pp. 13–31.

[84]   Karen Simonyan and Andrew Zisserman. *Very Deep Convolutional Networks
        for Large-Scale Image Recognition*. 2014. URL:
        https://arxiv.org/abs/1409.1556.

[85]   ASHBINDU SINGH. "Review Article Digital change detection techniques
        using remotely-sensed data". In: *International Journal of Remote Sensing* 10.6
        (1989), pp. 989–1003. URL: https://doi.org/10.1080/01431168908903939.

[86]   G M Smith. "The development of integrated object-based analysis of EO
        data within UK national land cover products". In: *Lecture Notes in
        Geoinformation and Cartography*. Lecture notes in geoinformation and
        cartography. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008,
        pp. 513–528.

[87]   Mark Spalding. *World atlas of mangroves*. Routledge, 2010.

[88]   Carole H Sudre et al. "Generalised dice overlap as a deep learning loss
        function for highly unbalanced segmentations". In: *Deep learning in medical
        image analysis and multimodal learning for clinical decision support*. Springer,
        2017, pp. 240–248.

[89] Christian Szegedy et al. "Going Deeper With Convolutions". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015.

[90] Daniel Tomowski, Manfred Ehlers, and Sascha Klonus. "Colour and texture based change detection for urban disaster analysis". In: *2011 Joint Urban Remote Sensing Event*. Munich, Germany: IEEE, Apr. 2011.

[91] John R Townshend et al. "Global characterization and monitoring of forest cover using Landsat data: opportunities and challenges". In: *International Journal of Digital Earth* 5.5 (2012), pp. 373–397.

[92] J Trinder and M Salah. "Aerial images and LiDAR data fusion for disaster change detection". In: *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci* 1 (2012), pp. 227–232.

[93] C Unsalan and Kim L Boyer. "Linearized vegetation indices based on a formal statistical framework". In: *IEEE Transactions on geoscience and remote sensing* 42.7 (2004), pp. 1575–1585.

[94] Cemr Unsalan. "Detecting changes in multispectral satellite images using time dependent angle vegetation indices". In: *2007 3rd International Conference on Recent Advances in Space Technologies*. IEEE. 2007, pp. 345–348.

[95] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 1999.

[96] Eric Vermote et al. "Preliminary analysis of the performance of the Landsat 8/OLI land surface reflectance product". en. In: *Remote Sens. Environ.* 185.Iss 2 (Apr. 2016), pp. 46–56.

[97] Anna Versluis and John Rogan. "Mapping land-cover change in a Haitian watershed using a combined spectral mixture analysis and classification tree procedure". en. In: *Geocarto Int.* 25.2 (Apr. 2010), pp. 85–103.

[98] Volker Walter. "Object-based classification of remote sensing data for change detection". en. In: *ISPRS J. Photogramm. Remote Sens.* 58.3-4 (Jan. 2004), pp. 225–238.

[99] Prosper Washaya, Timo Balz, and Bahaa Mohamadi. "Coherence change-detection with sentinel-1 for natural and anthropogenic disaster monitoring in urban areas". In: *Remote Sensing* 10.7 (2018), p. 1026.

[100] Jinru Xue and Baofeng Su. "Significant Remote Sensing Vegetation Indices: A Review of Developments and Applications". In: *Journal of Sensors* 2017 (May 2017), p. 1353691. ISSN: 1687-725X. DOI: 10.1155/2017/1353691. URL: https://doi.org/10.1155/2017/1353691.

[101] Abyot Yismaw et al. "Forest cover change detection using remote sensing and GIS in Banja district, Amhara region, Ethiopia". In: *International Journal of Environmental Monitoring and Analysis* 2.6 (2014), p. 354.

[102] Weiqi Zhou, Austin Troy, and Morgan Grove. "Object-based land cover classification and change analysis in the Baltimore metropolitan area using multitemporal high resolution remote sensing data". en. In: *Sensors (Basel)* 8.3 (Mar. 2008), pp. 1613–1636.

[103] Annus Zulfiqar et al. "AI-ForestWatch: semantic segmentation based end-to-end framework for forest estimation and change detection using multi-spectral remote sensing imagery". In: *Journal of Applied Remote Sensing* 15.2 (2021), p. 024518.

[104]   Alain F Zuur, Elena N Ieno, and Graham M Smith. "Principal component analysis and redundancy analysis". In: *Analysing ecological data* (2007), pp. 193–224.